200x85 Web Application

Tournaments

Introduction:

Tournaments are events hosted by 200x85 team. Each tournament have it's own set of criterias based on birth year and skillset (division), each division got limited amount of teams. For an example division 'Boys', '2009', 'AAA' allowing only 10 teams per that specific division. Per tournament there are usually set of different divisions determined by tournament director.

User groups:

Admin Director Team Representative

- Admin: This user role can create, edit, delete, update tournament information. The default landing page for admin screen shoud provide information about an existing tournaments, information about overall registration success. From main landing screen admin also able to either create / edit tournament info or get more details about an existing events.



Wireframe for adding new or editing tournament info. Admin has an option to structure tournament via set of divisions than can have an individual set of parameters. For an example B.O. / Girls-Boys / Payement info and etc. Each division can be easily cloned and updated.

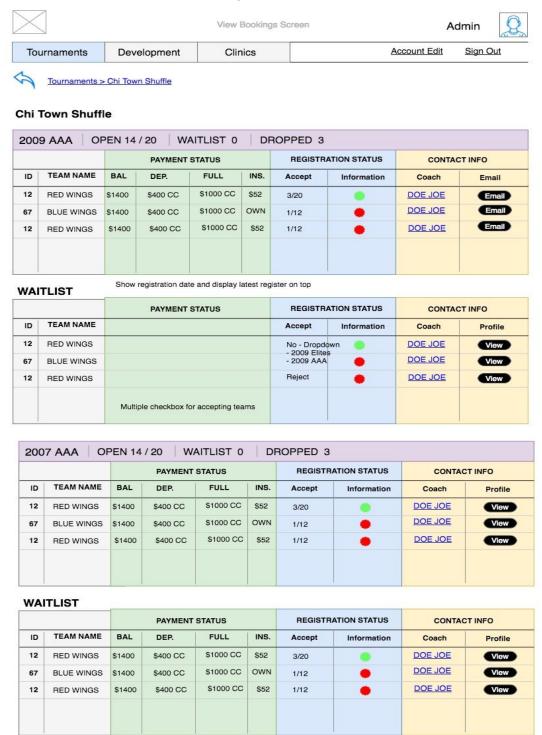
Adding New To		ing New Tournament Screen	A	Admin	
ournaments	Development	Clinics	Account Edit	Sign Out	
Tournaments > A	dd New Tournament				
Title		Long description	on	j	
Description					
Location / City / State		Registration C	Open Date End Date		
Add Division					
Division Title		Entry Fee			
2009 ELITE AA	A Boys	\$1,750	\$500		
Birth Year	Boys / Girls	Early Bid	Early Bid E	xp.	
2009	Boys	\$150	05/31/17		
Limit per devi	sion Level	Multi Tear	m Canadian		
15	ELITE AAA	\$150	\$350		
25		Balance [Due Balance Du	ie Date	
		\$1,250	09/31/17		
		Balance 1	Team Disc Balance Ca	nada	
		\$1,150	\$900		
		\$1,150	\$900		

Enter a title for the division such as Men's Singles, Over 30 Co-ed, Under 18, etc. Each tournament needs at least one division. Multiple divisions can be added.

Add Division

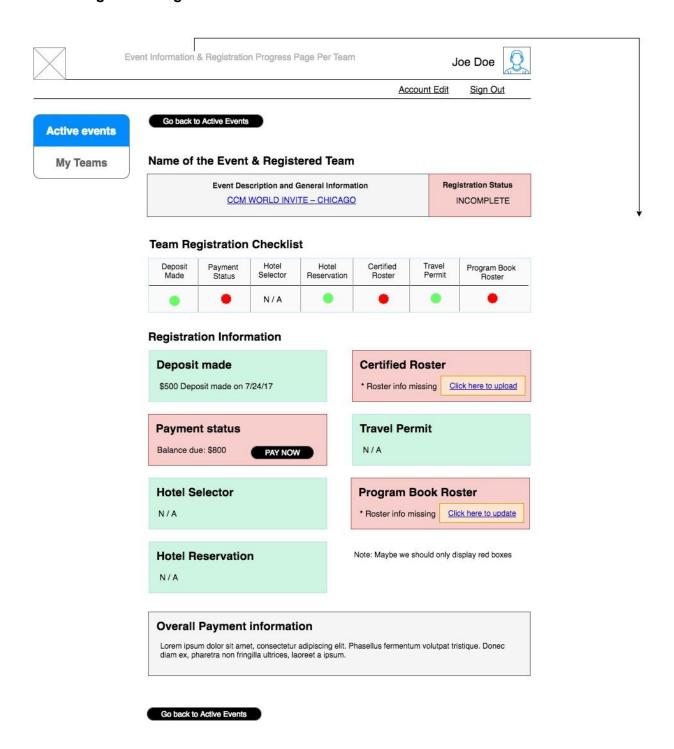
CLONE

Wireframe for viewing single tournament info. Admin can see information related to single event, can approve teams and move them from waiting list to main division table. Also from that screen admin go one level deeper and get access to the following screens: Team Respresentative information and Page where all team required assetes should be stored.



Admin should have an option to track team registration success and also button to sends an auto email, letting team representative know that some information are still required. In that auto email sent team rep. should get list of missing information to provide and link to the page where he/she can upload or add missing info or payment. Red dot's for data that is still missing.

Admin single team registration success wireframe:



- Team representative: This user role have to sign up for an account prior ability to register for tournaments. TR has an option to add multiple team profiles and then register team(s) for tournaments. The main TR resposibility to provide information and payemnts requested by admin.

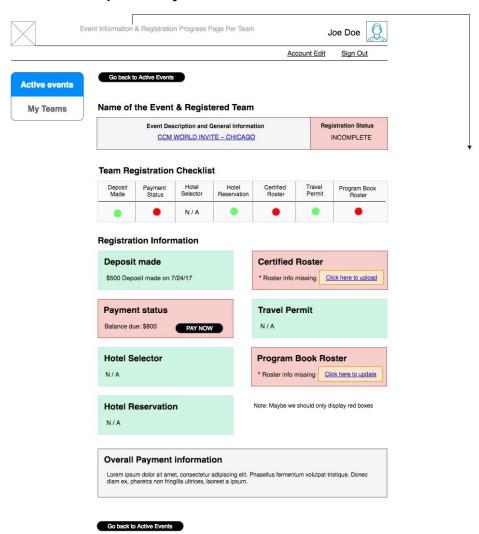
Team representative infromation wireframe: This information TR needs to provide in ordert to signup for an account and also edit in the future.

		Account Edit Sign C
Active events	Team Representative Inf	ormation
My Teams	Team Rep. Name	Team Rep. Phone #1
	Team Rep Email	Team Rep. Phone #2
	Organization Info	
	Organization Name	Billing Address
	Website URL	City
	Facebook	State ZipCode
	Instagram	Country
	Twitter	

TR main landing page wireframe: After registration of teams TR should see all active events and title of team that were registered along with overall status of registration.

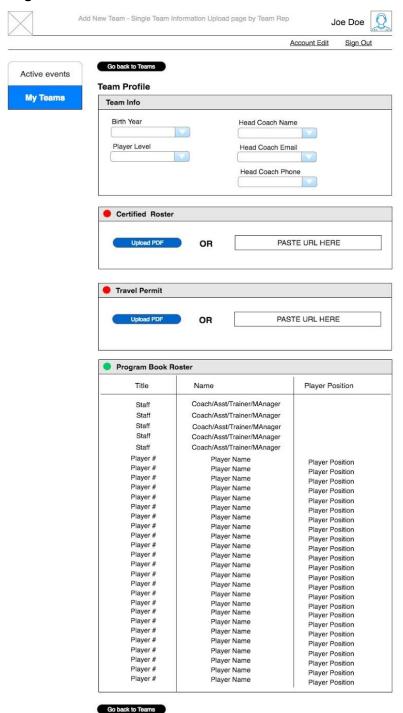


Information to provide by TR wireframe:



All checkpoint are connected with **Single Team infromation wireframe.** For an example if Certfied Roster Dot is red TR can go to team page and upload PDF.

Single Team information wireframe:



Note: An exact set of fields to be confirmed with 200x85 team.

- Director: This user role is similar to Admin. The only exception we not providing payment information and instead displaying to 2 more fields pulled from