

Tournament Management Software

UA Badminton Club

Status Update 2

Junfeng Xu, Enfa George, Rupal Jain, Urvika Gola



Structure of the Presentation

1. Plan & Accomplishment
2. Current Status
3. System Description
4. Challenges
5. Project Management
6. Demo

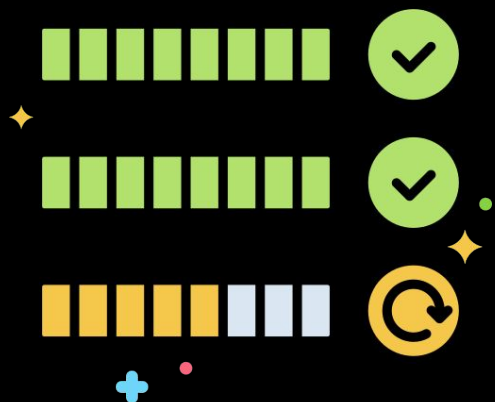


Plan & Accomplishment

Plan and Accomplishment

- Plan for this iteration
 - Backend support for the user login/signup functions
 - Database setup and database table creation
 - Player dashboard functionalities
 - Front end UI web pages
 - Backend support
 - Store and retrieve player profile
 - Store and retrieve player event registration
 - Testing for completed functionalities
 - Customer showcase
- Team accomplishment

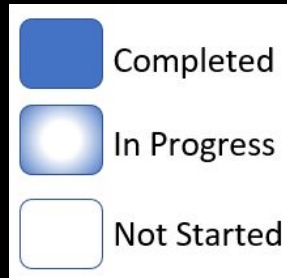




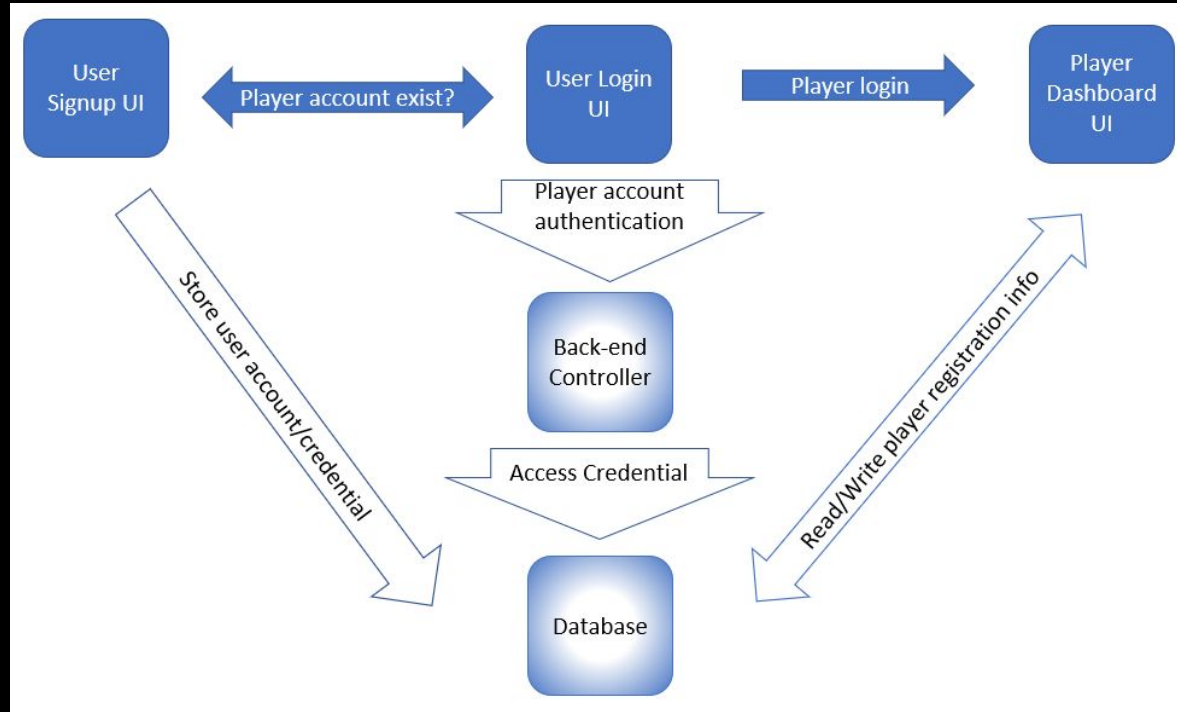
Current
Status

Current Status with Block Diagrams

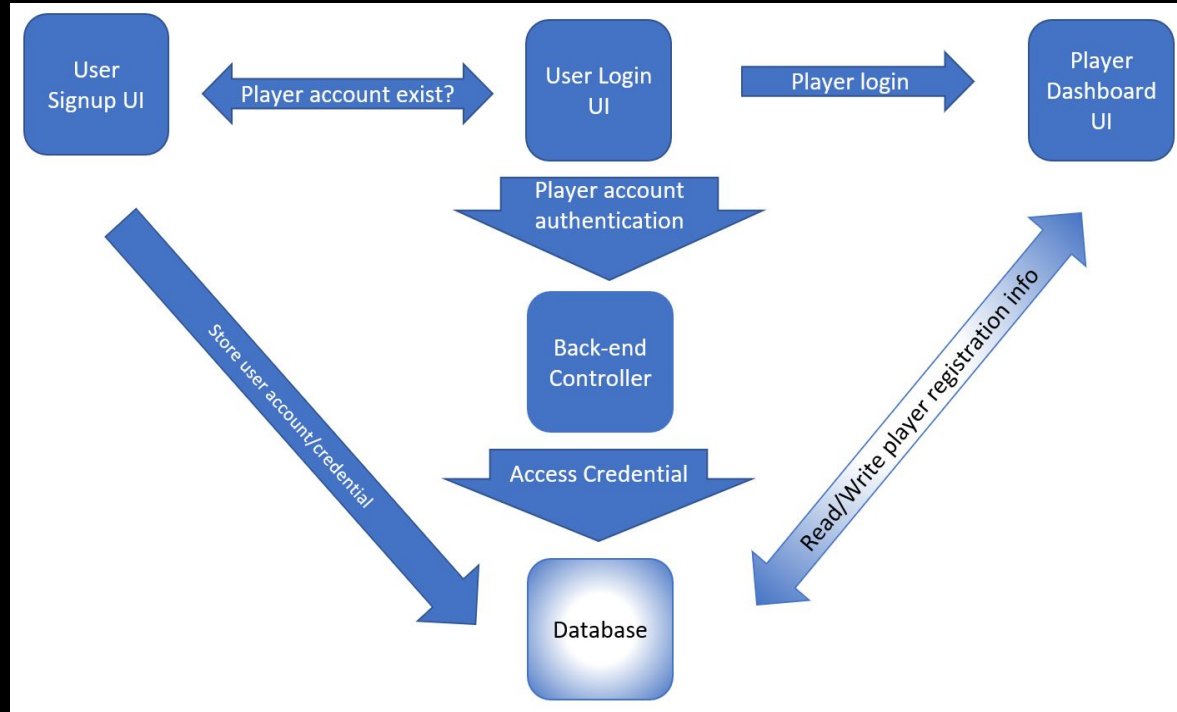
- To show our progress since last iteration
- Module by module
- Color code



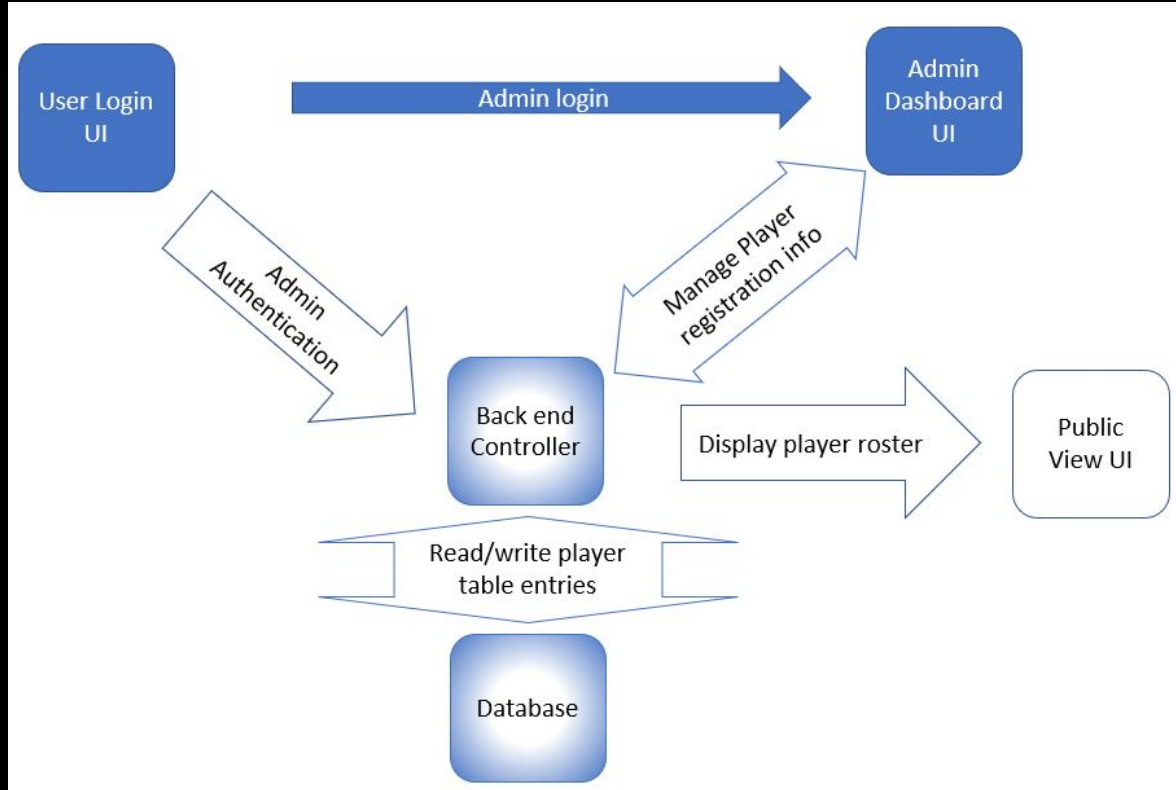
Registration module (Status 1 condition)



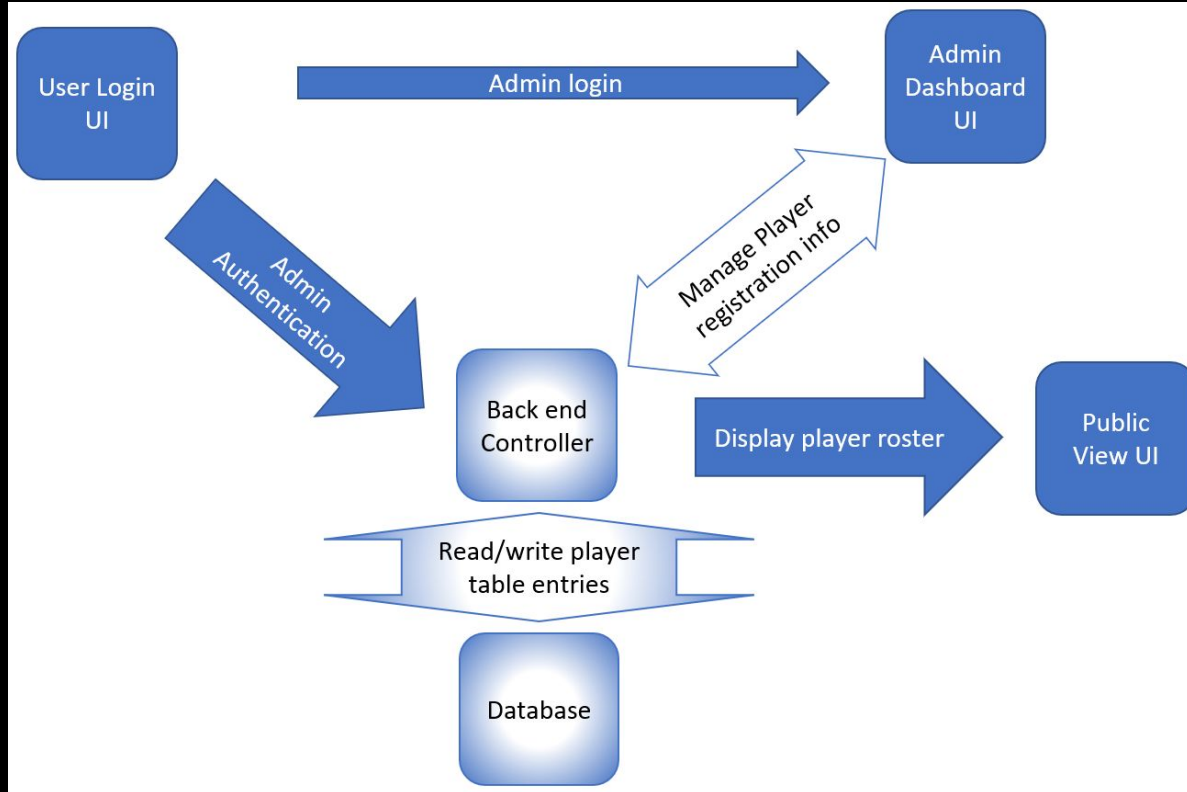
Registration module (Status 2 condition)



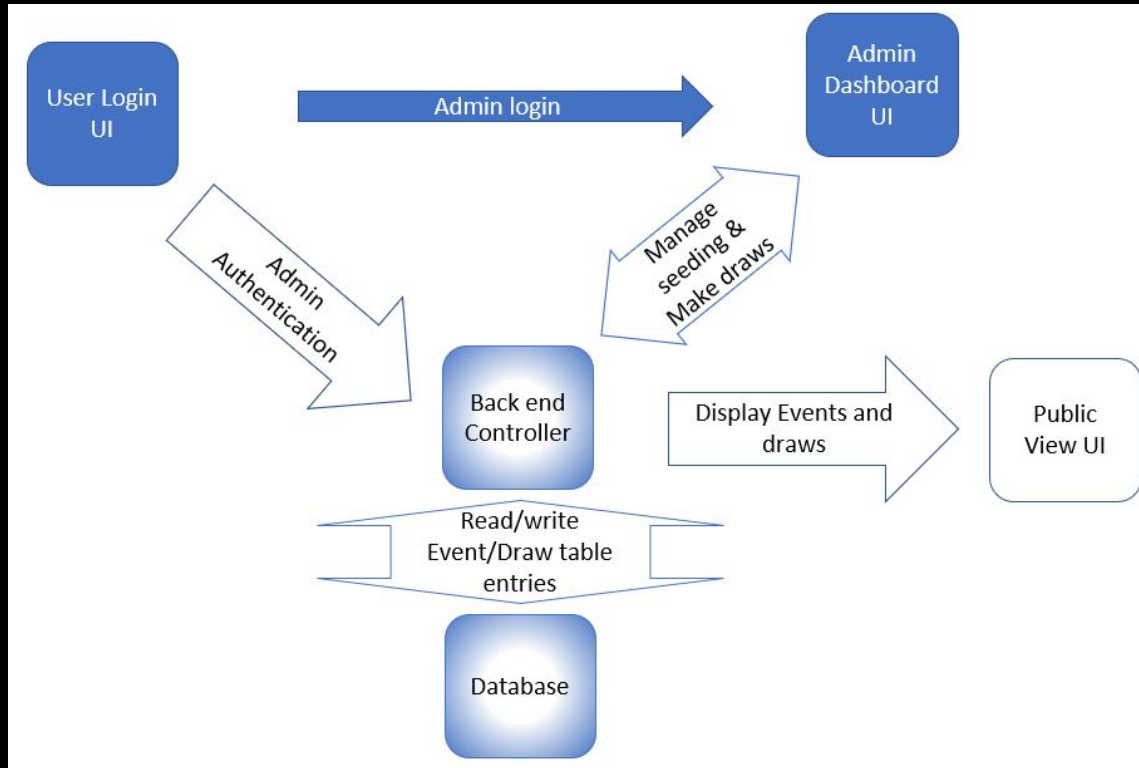
User Management module (Status 1 condition)



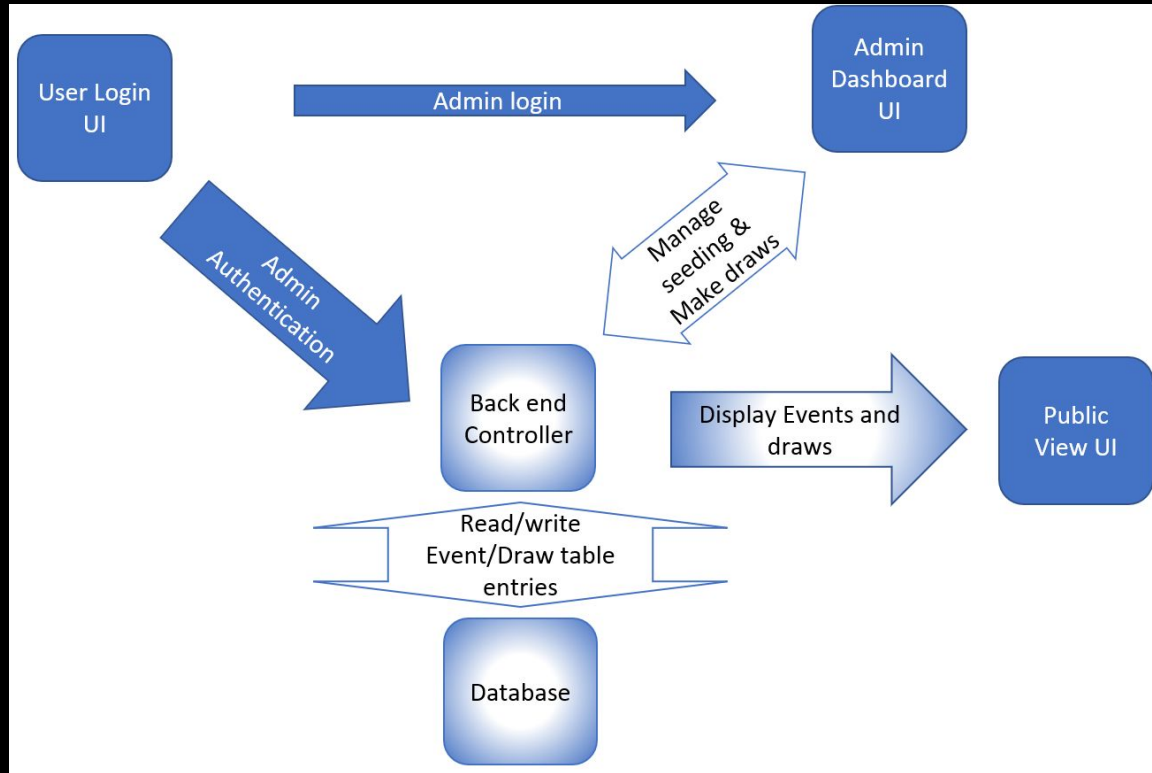
User Management module (Status 2 condition)



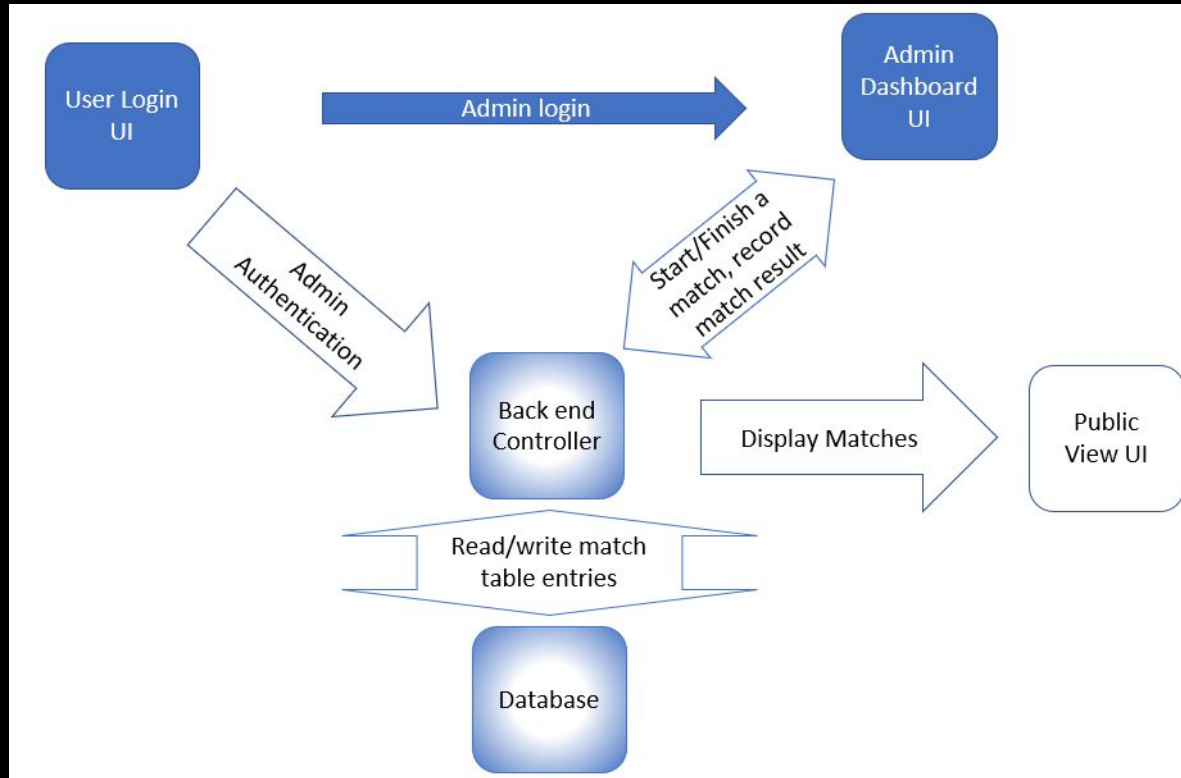
Seed & Draw module (Status 1 condition)



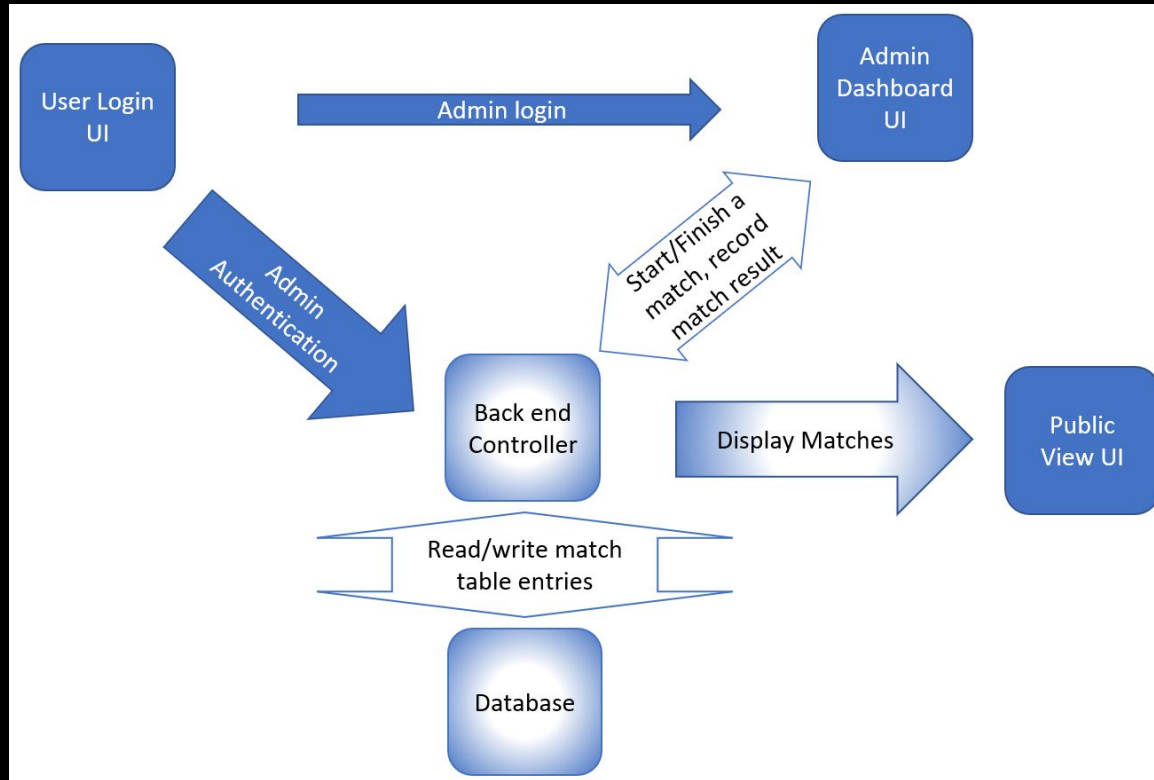
Seed & Draw module (Status 2 condition)



Match Management module (Status 1 condition)



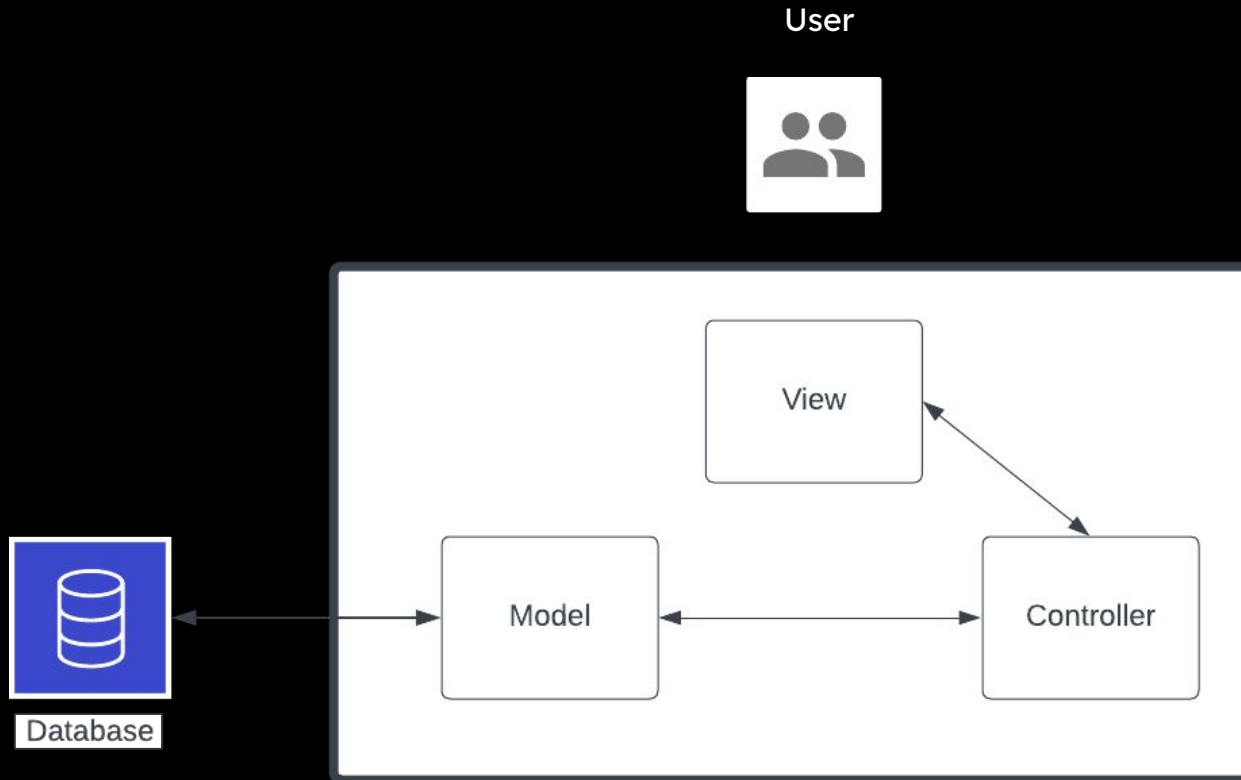
Match Management module (Status 2 condition)



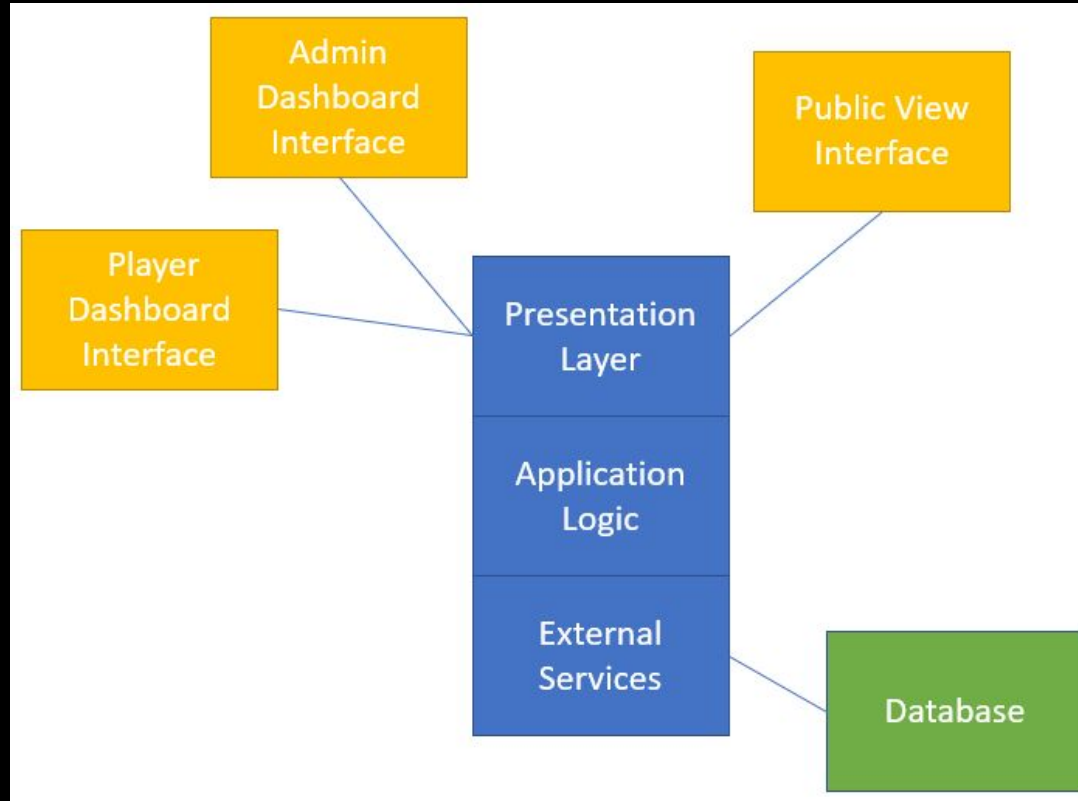


System Description

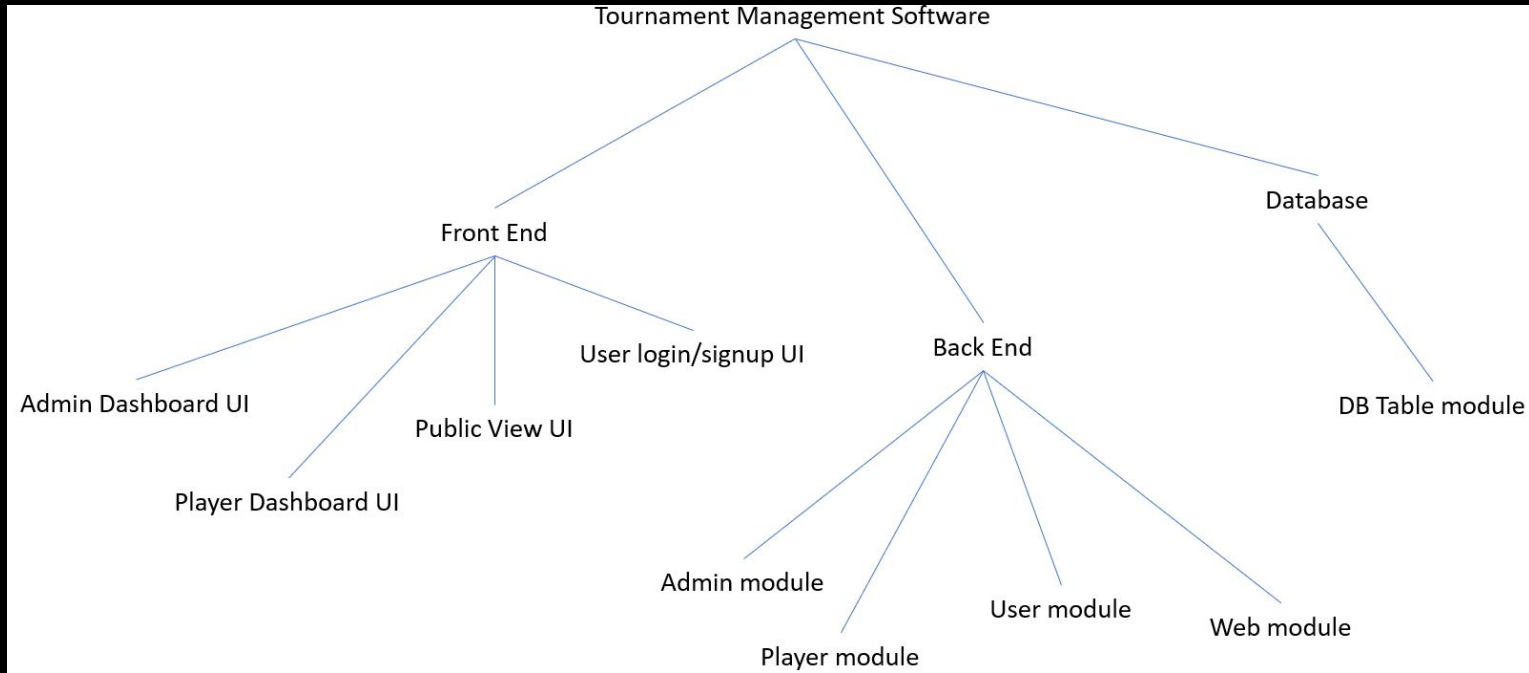
System Overview



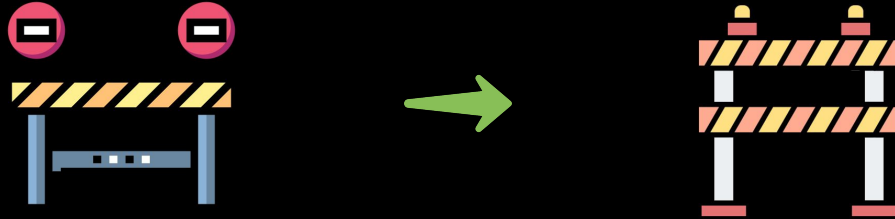
System Architecture Views - Logical View



System Architecture Views - Development View



Challenges



Overcoming Database Hurdles

- **Challenge** - Difficulty starting Oracle on Amazon EC2 instance due to configuration issues after installation.
- **Attempted Solution:** Reinstalling Oracle DB on Amazon EC2, but issue persisted.
- **Impact:** Delayed timeline by 3 days, resulting in increased development time.
- **Solution:** Leveraged alternative database system - Postgres. Installed and familiarized with Postgres on Mac locally,
- **Next Steps:** Installed **Postgres on Amazon RDS** and successfully connected it with **EC2**, resolving the Oracle configuration challenges.
- **Results:** Connected the product database setup using **SQLAlchemy** to our **Flask** Application, and the project timeline was back on track.

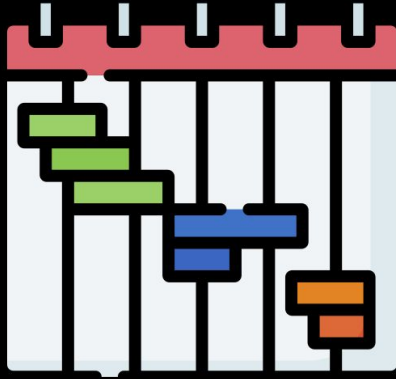
A retrospective...

- Key takeaways:
Flexibility and **adaptability** are crucial in overcoming unforeseen challenges.
- How?
By being open to alternative solutions and willing to adapt to changing circumstances, we were able to overcome obstacles and ultimately achieve our goals.

Benefits of Using Amazon RDS for our Software

The screenshot displays the Amazon RDS console interface. The top navigation bar includes the AWS logo, a 'Services' menu, a search bar, and regional/user information. The left sidebar lists navigation options for Amazon RDS, with 'Databases' highlighted. The main content area shows the breadcrumb path 'RDS > Databases > uabadmintournament' and the instance name 'uabadmintournament' with 'Modify' and 'Actions' buttons. Below this is a 'Summary' section containing a table of instance details.

DB Identifier	CPU
uabadmintournament	<div><div></div></div> 4.64%
Status	Class
✓ Available	db.t3.micro
Role	Current activity
Instance	<div><div></div></div> 3 Connections
Engine	Region & AZ
PostgreSQL	us-east-2a



Project Management

Jira Sprint










- Sprint Duration - 10/Apr/23 to 25/Apr/23
-
- 3 Stories
- 40 Subtasks

Status Report

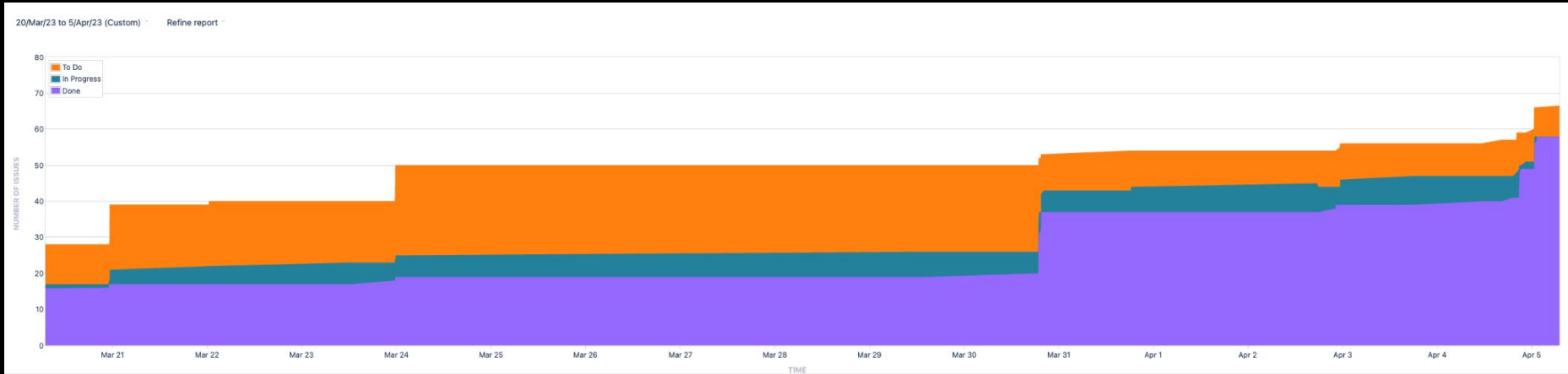
* Issue added to sprint after start time

Completed Issues


[View in Issue Navigator](#)

Key	Summary	Issue Type	Priority	Status	Story Points (- → 8)
UBT-31	Create Dashboard for Admin and Player	 Story	 Medium	 DONE	- → 2
UBT-37 *	Database Setup - 2	 Story	 High	 DONE	- → 3
UBT-40 *	Public View Web Pages	 Story	 Medium	 DONE	- → 3

Cumulative Flow Diagram



Incorporated 1, 2, 3 Story Points

Projects /  ua-badminton-tournam... /  UBT-31

Create Dashboard for Admin and Player

Attach

Create subtask

Link issue

⌵

⋮

Description

Add a description...

Subtasks

100% Done

UBT-33

Backend : Admin dashboard home page

✓

⌵

RJ

DONE

UBT-34

Backend : User dashboard HomePage

✓

⌵

JX

DONE

UBT-63

Backend: Player dashboard receiving data from frontend

✓

⌵

JX

DONE

UBT-64

Frontend: player dashboard receiving data from backend and display

✓

⌵

JX

DONE

UBT-66

Admin Dashboard Tournament Creation Button connection to backend

✓

⌵

RJ

DONE

UBT-67

Admin Dashboard Events and matches page receiving data from backend

✓

⌵

RJ

DONE

Done

✓ Done

⚡ Actions

Details

Assignee

RJ

Rupal Jain

Assign to me

Reporter

JX

Junfeng Xu

Labels

FrontEnd

Story Points

2

Sprint

None

+1

Priority

⌵

Medium

More fields

Original estimate

0m

Incorporated Story Points

UBT-38	Create, Configure Amazon EC2 Instance	UG	DONE
UBT-39	Install Oracle Server on EC2 Instance	UG	DONE
UBT-41	Code the 60% remaining tables in SQL Alchemy (Tournament, Events etc.)	UG	DONE
UBT-42	Correct the existing tables, due to no admin sign up and other fixes.	UG	DONE
UBT-49	Understand Oracle Server, Check IP and Port - BLOCKED	UG	DONE
UBT-50	Create User table in the PostgreSQL database	UG	DONE
UBT-51	Create Admin table in the PostgreSQL database	UG	DONE
UBT-52	Create Permission table in the PostgreSQL database	UG	DONE
UBT-53	Create Login table in the PostgreSQL database	UG	DONE
UBT-54	Create Tournament table in the PostgreSQL database	UG	DONE
UBT-55	Create Event table in the PostgreSQL database	UG	DONE
UBT-56	Create Result table in the PostgreSQL database	UG	DONE
UBT-57	Create Matches table in the PostgreSQL database	UG	DONE
UBT-58	Test the Admin, Player, Permissions tables by adding dummy data, verifying Primary Key and Foreign Key constraints using SQLAlchemy...	UG	DONE
UBT-59	Install and Run PostgreSQL on Mac (localhost)	UG	DONE
UBT-60	Create Amazon DBS with PostgreSQL and connect it to EC2	UG	DONE
UBT-65	Insert Admin and users	UG	DONE
UBT-70	[Database Backend Merge] : Signin	UG	DONE
UBT-71	[Database Backend Merge] Signup	UG	DONE
UBT-72	[Database Backend Merge] Player dashboard info loading	UG	DONE
UBT-74	User to users db temp changes	UG	DONE

Done

Done

Actions

Details

Assignee UG Urvika Gola

Reporter UG Urvika Gola

Labels Backend

Story Points 3

Sprint None +1

Priority High

More fields

Original estimate 0m

Time tracking 1w 7h 39m logged 3h 6m remaining

☒ Include subtasks

Epic Link None

Components None

Fix versions None

Created March 20, 2023 at 11:17 PM

Updated 12 hours ago

Resolved 12 hours ago

Configure

Daily Dashboard

Tuesday <April 4> (Sprint 2 Scrum 3)

1	Name	What did I work on yesterday 🤖	What am I working on today? 🤖	What issues are blocking me? 🤖
2	@Urvika Gola	<ul style="list-style-type: none"> Resolved EC2 DB issue by terminating the old instance, starting afresh with Amazon DBS with postgresSQL and connecting this to EC2 instance. UBT-60: Create Amazon DBS with PostgreSQL and connect it to EC2 DONE Configured for public access. Checked public access thorough my local machine. UBT-59: Install and Run PostgreSQL on Mac (localhost) DONE 	<ul style="list-style-type: none"> connected psql on Amazon DBS with SQL Alchemy ua-badminton-tournament: connected psql on Amazon DBS with SQL Alchemy CLOSED https://ugola.atlassian.net/browse/UBT-60 Report and PPT discussion, I am finishing the report and presentation for Status report 2. Reviewed Rupals PR ua-badminton-tournament: Added backend code for Admin Dashboard CLOSED 	
3	@Enfa Rose George	<ul style="list-style-type: none"> Connected the PostgreSQL db set up by Urvika to the project via SQL alchemy so that db object is available throughout the app. inserted init table value needs such as admin login via psq Wrote db operations for signup, loading player dashboard details Ensured that dashboards cannot be accessed by anyone outside of the soleowner using python decorator. Reviewed PR by team mates and recommended corrections/best practises before PR merge 	<ul style="list-style-type: none"> Wrote db operations for signup,, and updating almost all player details Ensured that dashboards cannot be accessed by anyone outside of the soleowner using python decorator. Implemented the logout functionality Reviewed PR by team mates and recommended corrections/best practises before PR merge 	<ul style="list-style-type: none"> [SOLVED] Signup sheet wasn't collecting all details needed to create a user in db. This was due to communication gap and was solved in the meeting.
4	@Junfeng Xu	<ul style="list-style-type: none"> I completed integrating the user login and sign up into the home page. ua-badminton-tournament: Homepage signin CLOSED I completed implementing the following functionalities to the player dashboard front end UI <ul style="list-style-type: none"> reading backend variables and displaying them on the front end UI for player dashboard. The inputs to the player dashboard front end UI are also able to be passed to backend controller UBT-63: Backend: Player dashboard receiving data from frontend DONE 	<ul style="list-style-type: none"> Working on status update 2 Created Architectural Diagrams organizing how to split the presentation and report to each team member preparing on my assignment of the presentation writing my assigned sections of the report Incorporated new changes by middleware: ua-badminton-tournament: signup page, player dashboard page are changed. CLOSED 	
5	@Rupal Jain	<p>I was working on the Admin dashboard Events page so that the events page should receive data from backend. UBT-66: Admin Dashboard Tournament Creation Button connection to backend DONE</p>	<ol style="list-style-type: none"> I am finishing the report and presentation for Status report 2. I will be concluding the backend connection for all the pages of Admin Dashboard 	<p>I have to reformat the data received from backend in necessary format readable to HTML. ua-badminton-tournament: Added connection for Admin dashboard for pages: Home, Tournament, Matches, Events OPEN</p>

