MIDDLE EAST TECHNICAL UNIVERSITY
NORTHERN CYPRUS CAMPUS

CNG 495

Fall – 2025

Capstone Progress Report

Name: Nurlan

Surname: İldırımlı

SID: 2745438

Project Title: University Sports Center Reservation System

Date of Submission: 01.12.2025

Contents

1 Project Description

This project proposes a cloud-based reservation platform for the University Sports Center. Students and staff can browse facilities (Football Pitch, Tennis Court, volleyball / Basketball court outdoor, Table Tennis), view available time slots, and create/cancel reservations. Authorized personnel can define working hours, capacities, maintenance windows, and policies (e.g., per-week booking limits).

The system will provide real-time availability, double-booking prevention. The web application will be implemented with a modern front-end and a serverless backend on Firebase. Data will be stored in a cloud database with transactional safeguards to ensure consistency during peak demand.

2 Cloud Delivery Models

2.1 SaaS (Software as a Service)

Users access the reservation web app to sign in, view facilities/slots, and create/cancel reservations. Staff use an admin dashboard to manage facilities, capacities, and policies.

2.2 PaaS (Platform as a Service)

- **Firebase Authentication** for secure login (email/school account).
- **Cloud Functions** to enforce these rules on the server side:
  - Check availability and prevent double booking.
  - Enforce policies (per-user weekly limits(if applicable), cancellation window).
  - Send e-mail notifications when enabled.

2.3 IaaS (Infrastructure as a Service)

Cloud Firestore will be my realtime database for storing facilities, slots, and reservations. Managed, scalable, and fault-tolerant infrastructure.

3 Diagrams

3.1 Use Case Diagram

Here you can see the use case diagram in Figure 1.

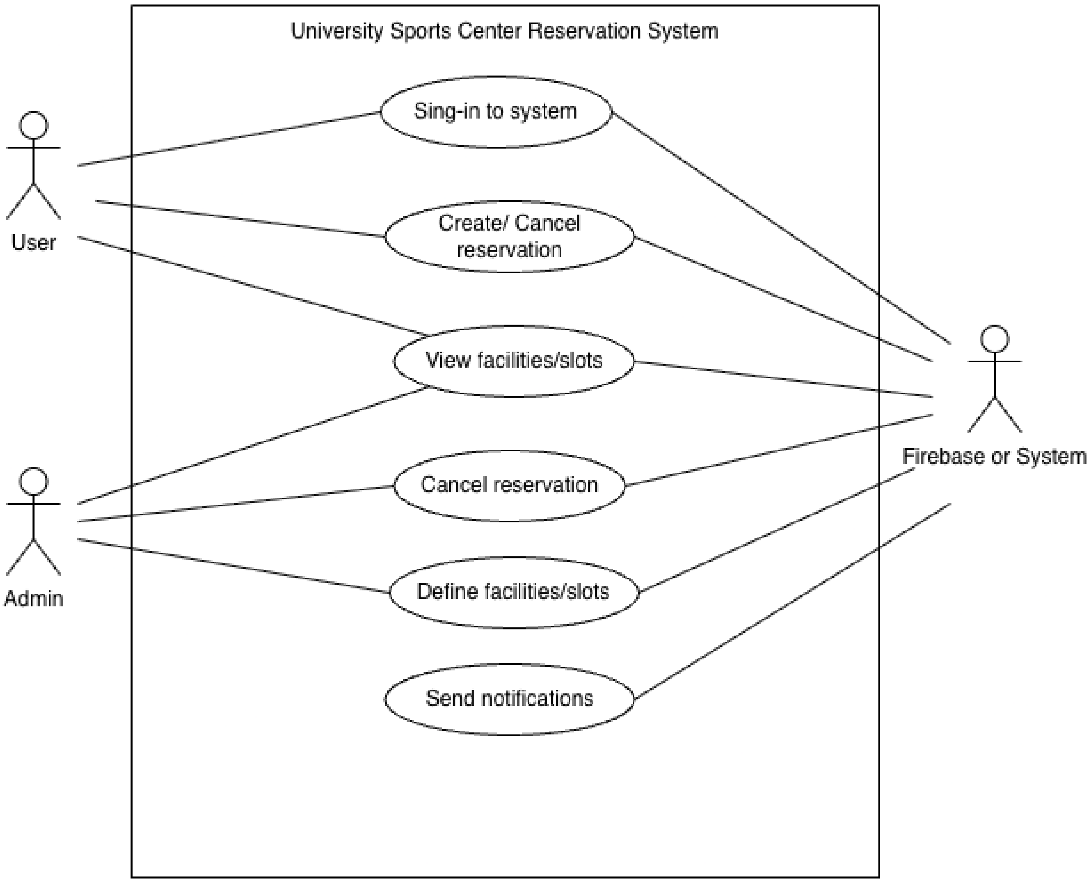


Figure 1: Use case diagram.

3.2 Data Flow Diagram

Here you can see the data flow diagrams in Figure 2 and Figure 3.

Context Level:

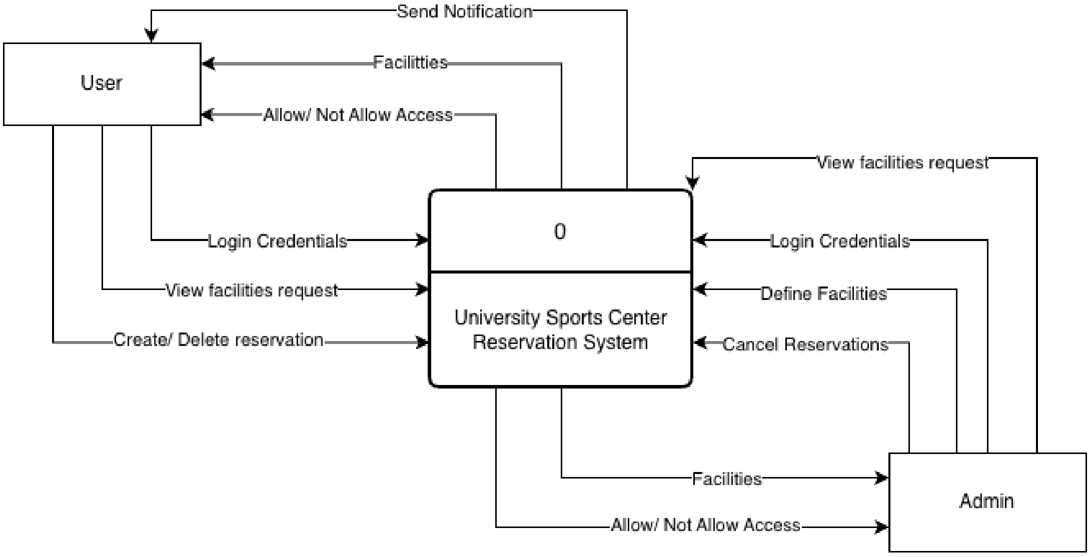


Figure 2: Context Level.

Level 0:

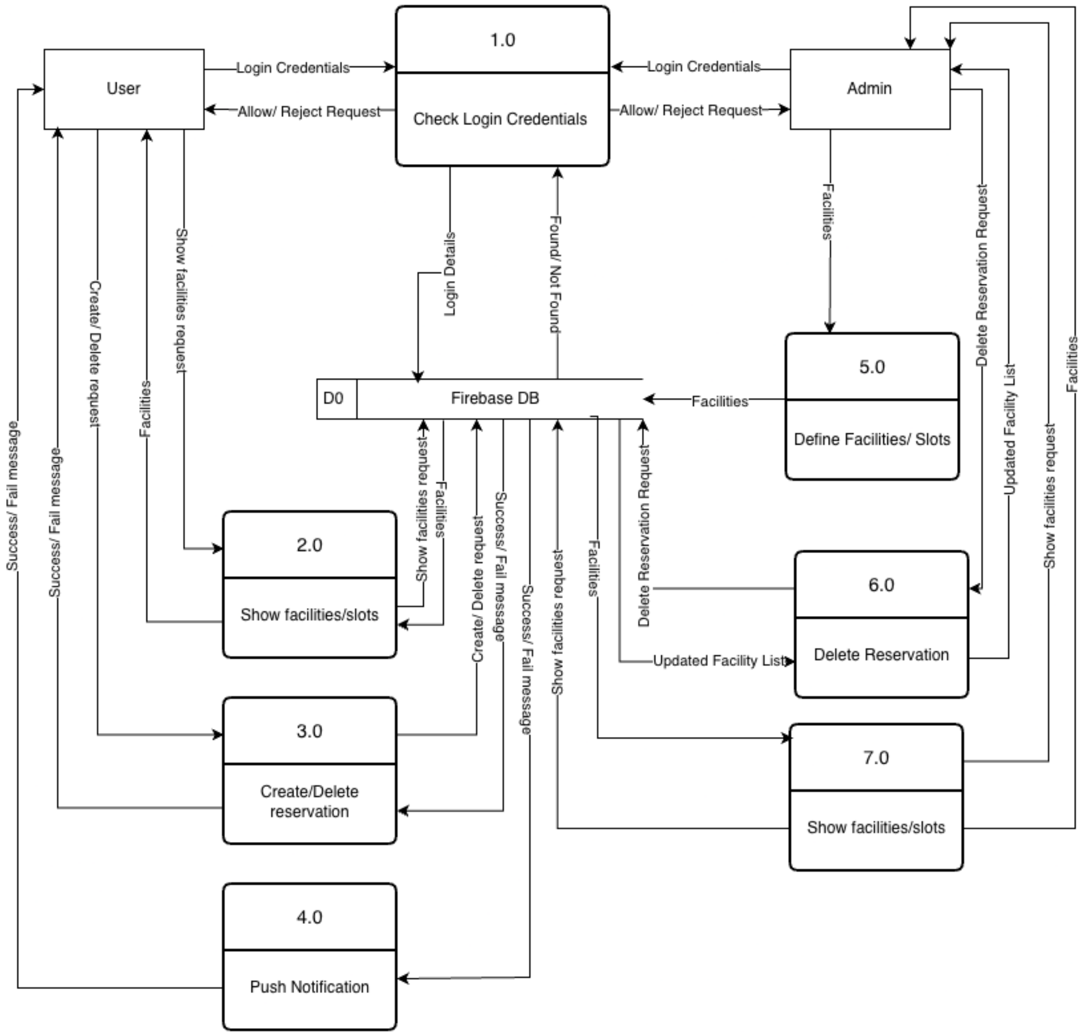


Figure 3: Level 0.

4 Data Types

- **Text:** Facility name, description, role, policy, username/email, reservation ID.
- **Numeric:** Capacity, slot start/end times, cancellation window (minutes/hours).

5 Computation

- **Authentication & Authorization:** Firebase Auth (role-based access).
- **Policy Enforcement:** Weekly per-user limits, cancellation window, no-show flagging.
- **Notifications:** The system will send confirmations and cancellation notifications through e-mail or push notifications. Notification delivery is handled automatically by Firebase Cloud Functions and Firebase Cloud Messaging.
- **Cancellation:** Cloud Functions system will enforce a cancellation window to prevent

last-minute cancellations that cause unused facility time. Each reservation will include a cancellation_window attribute (e.g., 1 hours before the reserved slot).

6 Technologies Tools

- **Front-end:** React (Vite) + TypeScript + Custom CSS.
- **Auth:** Firebase Authentication (Email/Password).
- **Database:** Cloud Firestore.
- **Backend:** Firebase Cloud Functions (Node.js + TypeScript).

7 Milestones Achieved

-Week of October 13–20

- Project initialized using React + Vite + TypeScript.
- Firebase project created and connected to the frontend.
- Firebase Authentication implemented (Email/Password login).

-Week of October 20–27

- Firestore structure designed for:
  - facilities
  - slots
  - reservations
- Facilities collection created and manually populated through Firebase Console.
- Facilities successfully fetched from Firestore and displayed on the frontend.
- Slots fetching implemented for selected facility.

-Week of October 27–November 3

- Student reservation creation flow completed.
- Slot availability updated in Firestore after reservation.
- “My Reservations” page implemented — students can list their own reservations.
- Reservation cancellation feature added for students.

-Week of November 3–10

- Admin panel structure created and restricted to admin users.
- Admin view implemented to display all reservations from Firestore.

-Week of November 10–17

- Admin actions implemented:
  - Mark reservation as completed
  - Mark reservation as not attended
- Cloud Functions implemented to send emails:
  - “You created a reservation”
  - “You did not attend the session…”
- Cloud Functions coded; deployment requires Firebase Blaze plan.

-Week of November 17–24

- SMTP configuration completed using Gmail App Password.
- Cloud Functions build completed locally.
- Project linting and TypeScript issues resolved for successful build.
- Deployment attempted; Firebase plan limitation identified and documented.

-Week of November 24–30

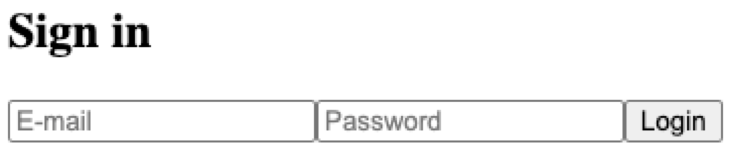
- Final integration of admin status change → email trigger logic (local environment).
- GitHub repository prepared with all source code.
- Progress Report content prepared (timeline, features, screenshots).

8 Work Completed by Nurlan İldırımlı

- Frontend environment setup
- Firebase Authentication integration
- Facilities and slots Firestore integration
- Reservation creation workflow

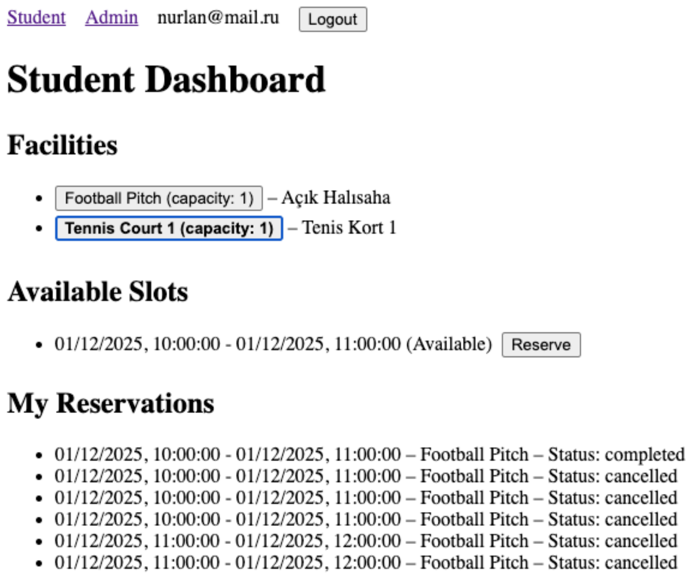
- Admin dashboard development
- Status update logic (“completed” / “not_attended”)
- Cloud Functions for email notifications
- Reservation ban logic

9 UI Screenshots



Sign in

Figure 4: Login Page: User signs in with email and password using Firebase Authentication.



Student Dashboard

Facilities

- Football Pitch (capacity: 1) – Açık Halısaha
- Tennis Court 1 (capacity: 1) – Tenis Kort 1

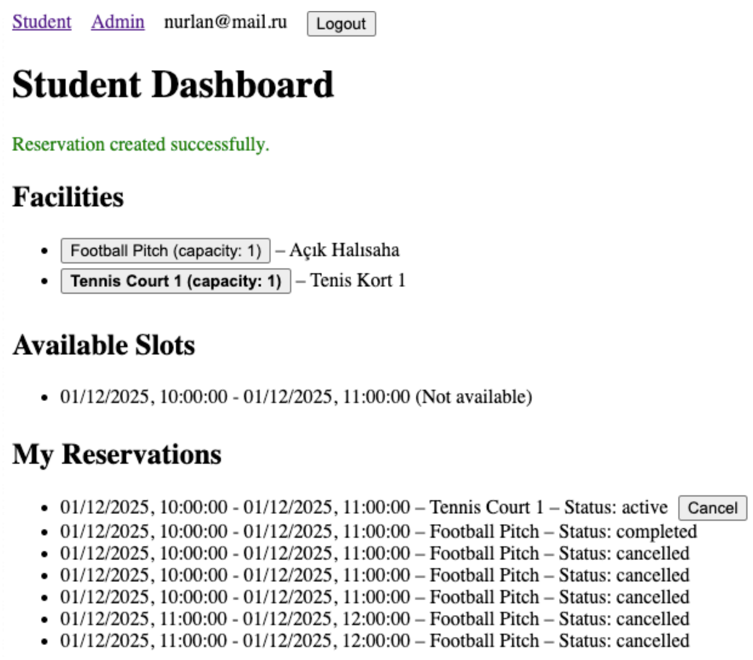
Available Slots

- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 (Available) Reserve

My Reservations

- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 – Football Pitch – Status: completed
- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 – Football Pitch – Status: cancelled
- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 – Football Pitch – Status: cancelled
- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 – Football Pitch – Status: cancelled
- 01/12/2025, 11:00:00 - 01/12/2025, 12:00:00 – Football Pitch – Status: cancelled
- 01/12/2025, 11:00:00 - 01/12/2025, 12:00:00 – Football Pitch – Status: cancelled

Figure 5: Student Dashboard: Student views all available facilities fetched from Firestore.



Student Admin nurlan@mail.ru Logout

Student Dashboard

Reservation created successfully.

Facilities

- Football Pitch (capacity: 1) – Açık Halısaha
- Tennis Court 1 (capacity: 1) – Tenis Kort 1

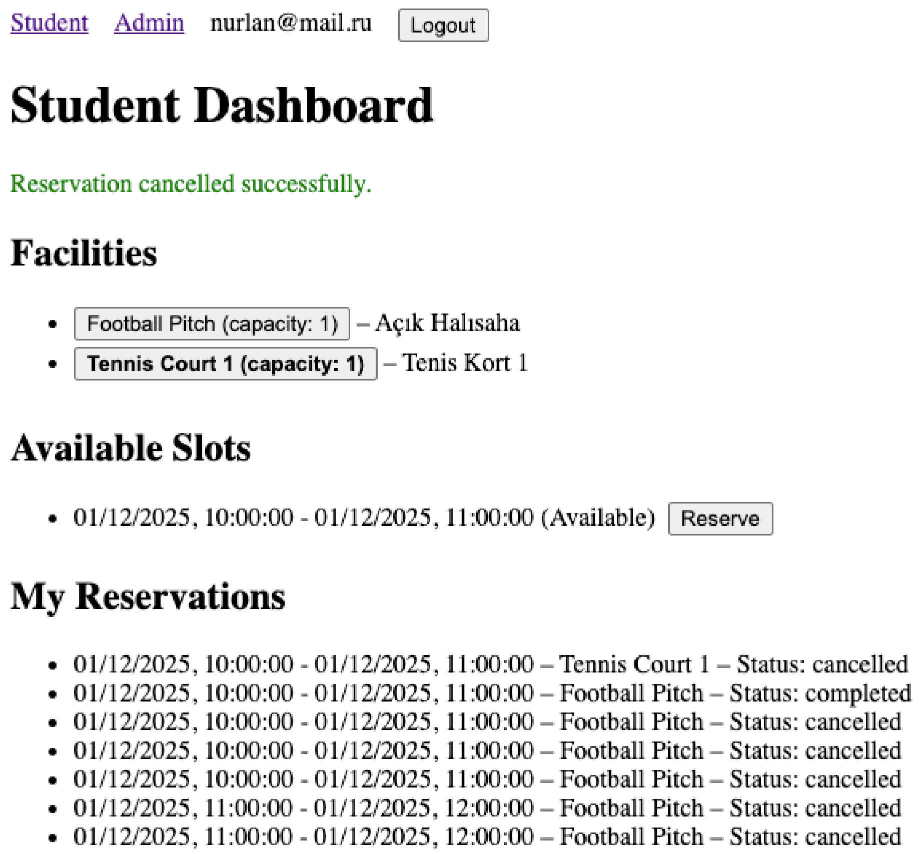
Available Slots

- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 (Not available)

My Reservations

- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 – Tennis Court 1 – Status: active Cancel
- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 – Football Pitch – Status: completed
- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 – Football Pitch – Status: cancelled
- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 – Football Pitch – Status: cancelled
- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 – Football Pitch – Status: cancelled
- 01/12/2025, 11:00:00 - 01/12/2025, 12:00:00 – Football Pitch – Status: cancelled
- 01/12/2025, 11:00:00 - 01/12/2025, 12:00:00 – Football Pitch – Status: cancelled

Figure 6: Reservation Taken: Student selects an available time slot and creates a reservation.



Student Admin nurlan@mail.ru Logout

Student Dashboard

Reservation cancelled successfully.

Facilities

- Football Pitch (capacity: 1) – Açık Halısaha
- Tennis Court 1 (capacity: 1) – Tenis Kort 1

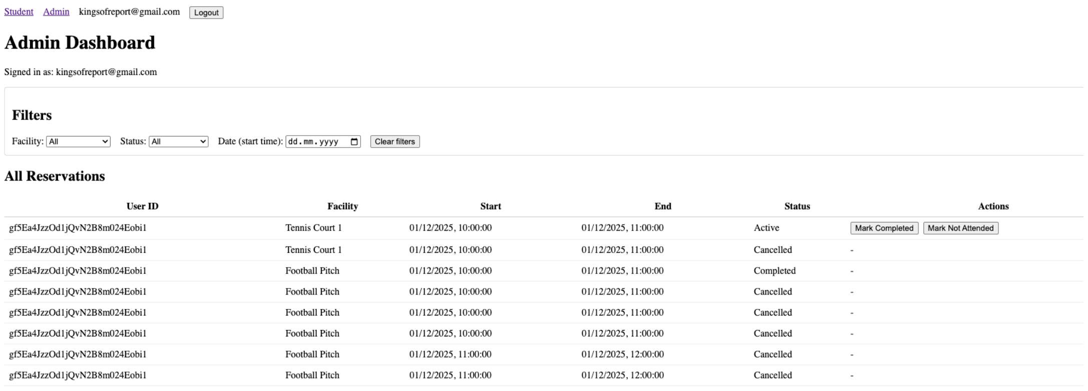
Available Slots

- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 (Available) Reserve

My Reservations

- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 – Tennis Court 1 – Status: cancelled
- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 – Football Pitch – Status: completed
- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 – Football Pitch – Status: cancelled
- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 – Football Pitch – Status: cancelled
- 01/12/2025, 10:00:00 - 01/12/2025, 11:00:00 – Football Pitch – Status: cancelled
- 01/12/2025, 11:00:00 - 01/12/2025, 12:00:00 – Football Pitch – Status: cancelled
- 01/12/2025, 11:00:00 - 01/12/2025, 12:00:00 – Football Pitch – Status: cancelled

Figure 7: Reservation Cancel: Student sees their own reservations and can cancel them.



Admin Dashboard

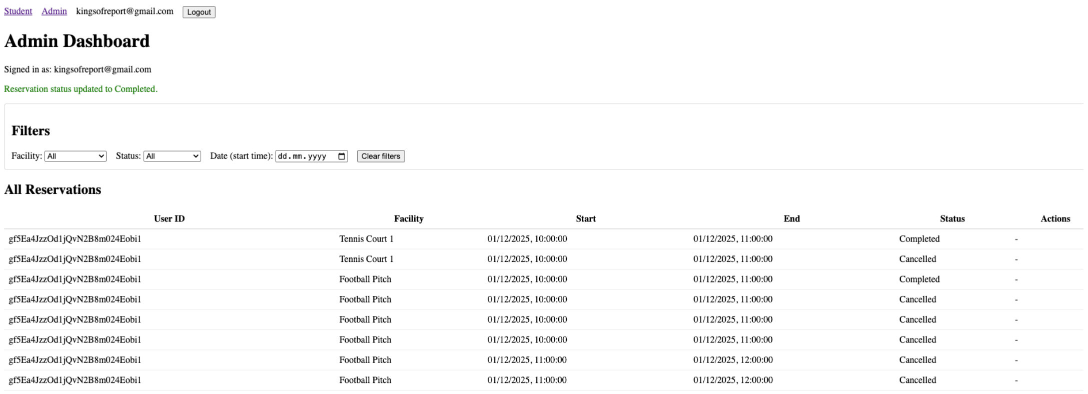
Signed in as: kingsofreport@gmail.com

Filters

All Reservations

User ID	Facility	Start	End	Status	Actions
gf5Ea4JzzOd1jQvN2B8m024Eobi1	Tennis Court 1	01/12/2025, 10:00:00	01/12/2025, 11:00:00	Active	Mark Completed Mark Not Attended
gf5Ea4JzzOd1jQvN2B8m024Eobi1	Tennis Court 1	01/12/2025, 10:00:00	01/12/2025, 11:00:00	Cancelled	-
gf5Ea4JzzOd1jQvN2B8m024Eobi1	Football Pitch	01/12/2025, 10:00:00	01/12/2025, 11:00:00	Completed	-
gf5Ea4JzzOd1jQvN2B8m024Eobi1	Football Pitch	01/12/2025, 10:00:00	01/12/2025, 11:00:00	Cancelled	-
gf5Ea4JzzOd1jQvN2B8m024Eobi1	Football Pitch	01/12/2025, 10:00:00	01/12/2025, 11:00:00	Cancelled	-
gf5Ea4JzzOd1jQvN2B8m024Eobi1	Football Pitch	01/12/2025, 10:00:00	01/12/2025, 11:00:00	Cancelled	-
gf5Ea4JzzOd1jQvN2B8m024Eobi1	Football Pitch	01/12/2025, 11:00:00	01/12/2025, 12:00:00	Cancelled	-
gf5Ea4JzzOd1jQvN2B8m024Eobi1	Football Pitch	01/12/2025, 11:00:00	01/12/2025, 12:00:00	Cancelled	-

Figure 8: Admin Dashboard: Admin views all reservations in the system, including details and status.



Admin Dashboard

Signed in as: kingsofreport@gmail.com

Reservation status updated to Completed.

Filters

All Reservations

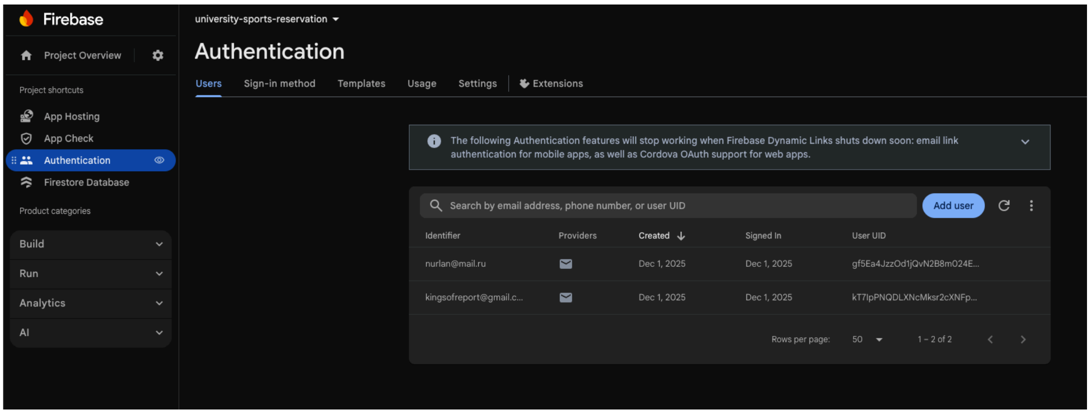
User ID	Facility	Start	End	Status	Actions
gf5Ea4JzzOd1jQvN2B8m024Eobi1	Tennis Court 1	01/12/2025, 10:00:00	01/12/2025, 11:00:00	Completed	-
gf5Ea4JzzOd1jQvN2B8m024Eobi1	Tennis Court 1	01/12/2025, 10:00:00	01/12/2025, 11:00:00	Cancelled	-
gf5Ea4JzzOd1jQvN2B8m024Eobi1	Football Pitch	01/12/2025, 10:00:00	01/12/2025, 11:00:00	Completed	-
gf5Ea4JzzOd1jQvN2B8m024Eobi1	Football Pitch	01/12/2025, 10:00:00	01/12/2025, 11:00:00	Cancelled	-
gf5Ea4JzzOd1jQvN2B8m024Eobi1	Football Pitch	01/12/2025, 10:00:00	01/12/2025, 11:00:00	Cancelled	-
gf5Ea4JzzOd1jQvN2B8m024Eobi1	Football Pitch	01/12/2025, 10:00:00	01/12/2025, 11:00:00	Cancelled	-
gf5Ea4JzzOd1jQvN2B8m024Eobi1	Football Pitch	01/12/2025, 11:00:00	01/12/2025, 12:00:00	Cancelled	-
gf5Ea4JzzOd1jQvN2B8m024Eobi1	Football Pitch	01/12/2025, 11:00:00	01/12/2025, 12:00:00	Cancelled	-

Figure 9: Admin Action: Admin updates a reservation's status (completed or not attended).

10 Cloud Technologies Tutorial

Firebase Authentication

Firebase Authentication provides a simple way to handle user login and identity management. In this project, Email/Password authentication is used. The frontend connects to Firebase Auth through the `getAuth()` method, allowing users to sign in securely. After login, Firebase returns a user token that is used to control access to student or admin pages.



The following Authentication features will stop working when Firebase Dynamic Links shuts down soon: email link authentication for mobile apps, as well as Cordova OAuth support for web apps.

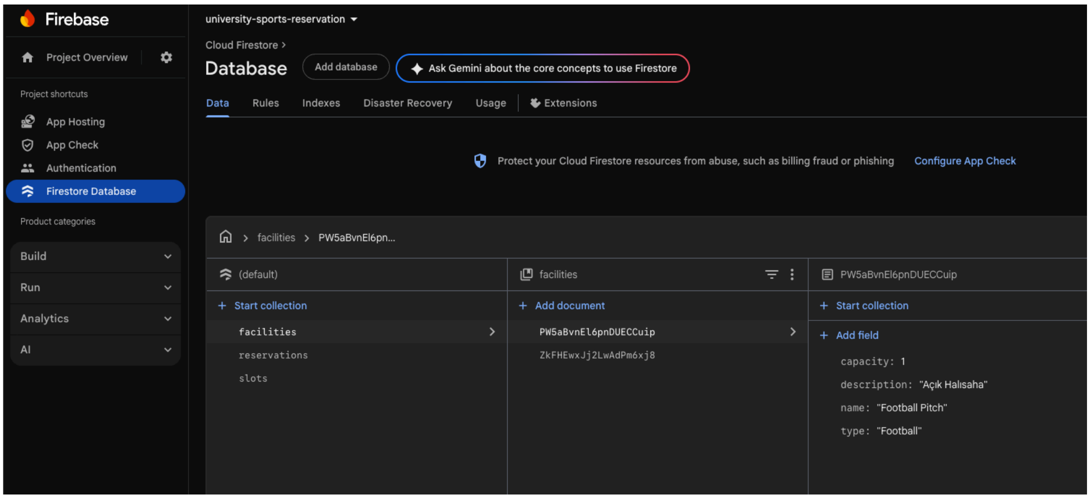
Search by email address, phone number, or user UID

Identifier	Providers	Created ↓	Signed In	User UID
nurlan@mail.ru	✉	Dec 1, 2025	Dec 1, 2025	gf5Ea4JzzOd1jQvN2B8m024E...
kingsofreport@gmail.c...	✉	Dec 1, 2025	Dec 1, 2025	kT7IpPNQDLXNcMksr2cXNFp...

Rows per page: 50 1 – 2 of 2

Cloud Firestore

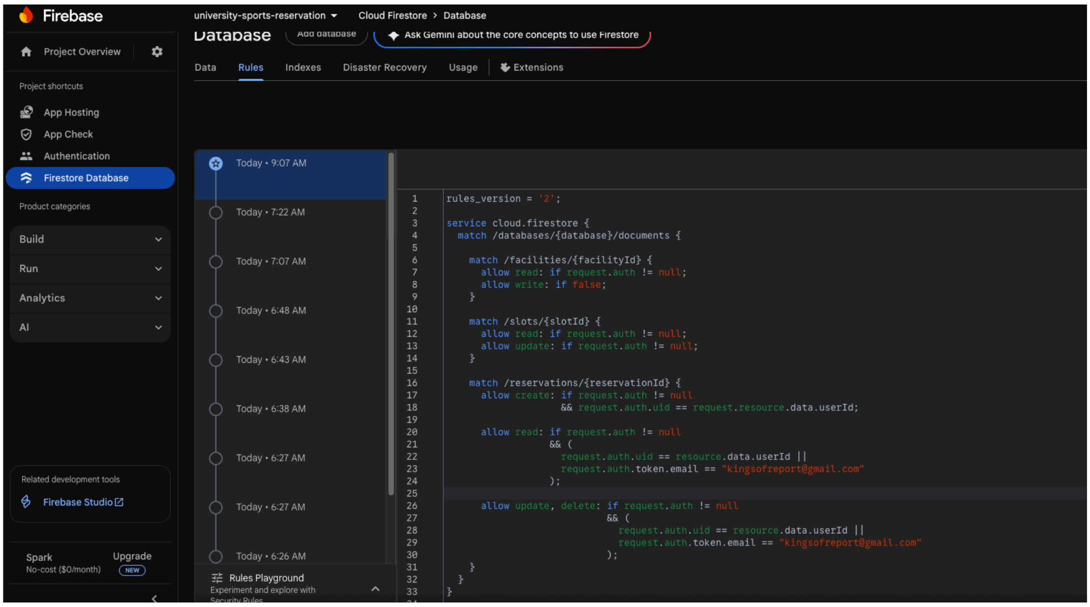
Cloud Firestore is a NoSQL document database used to store facilities, slots, and reservations. Data is organized into collections such as `facilities`, `slots`, and `reservations`. Queries fetch real-time updates, allowing the UI to instantly reflect changes (e.g., a slot becoming unavailable). Security Rules ensure that only authorized users can read or modify specific documents.



Protect your Cloud Firestore resources from abuse, such as billing fraud or phishing

facilities > PW5aBvnEl6pn...

(default)	facilities	PW5aBvnEl6pnDUECCuip
facilities	PW5aBvnEl6pnDUECCuip	capacity: 1
reservations	ZkFHEwxJj2LwAdPm6xj8	description: "Açık Halısaha"
slots		name: "Football Pitch"
		type: "Football"



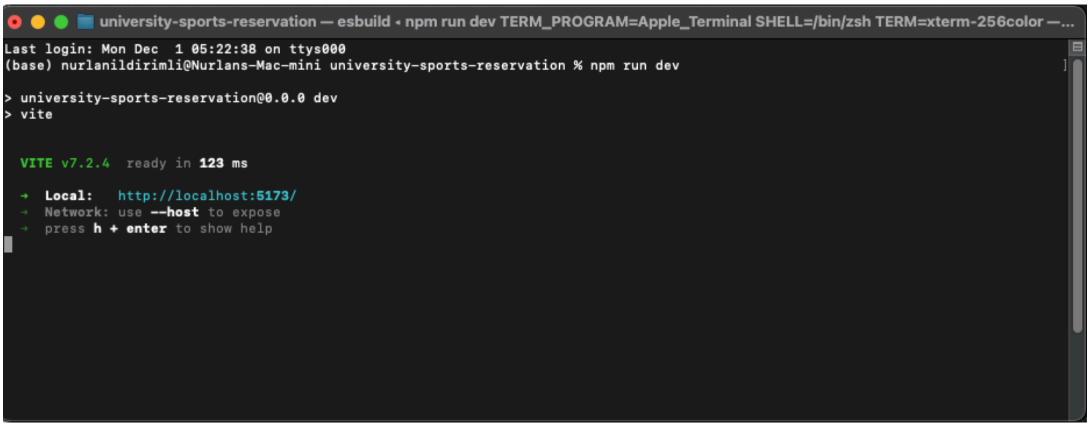
```
rules_version = '2';

service cloud.firestore {
  match /databases/{database}/documents {

    match /facilities/{facilityId} {
      allow read: if request.auth != null;
      allow write: if false;
    }

    match /slots/{slotId} {
      allow read: if request.auth != null;
      allow update: if request.auth != null;
    }

    match /reservations/{reservationId} {
      allow create: if request.auth != null
                    && request.auth.uid == request.resource.data.userId;

      allow read: if request.auth != null
                  && (
                    request.auth.uid == resource.data.userId ||
                    request.auth.token.email == "kingsofreport@gmail.com"
                  );

      allow update, delete: if request.auth != null
                            && (
                              request.auth.uid == resource.data.userId ||
                              request.auth.token.email == "kingsofreport@gmail.com"
                            );
    }
  }
}
```

Firebase Cloud Functions

Firebase Cloud Functions enable server-side logic to run automatically. In this project, functions are written in TypeScript and triggered by Firestore events. When a reservation is created or marked "not attended," a corresponding function sends an email to the student. These functions run on Google's serverless infrastructure, removing the need for a dedicated backend server.

Firebase Hosting

Firebase Hosting is used to serve the React application. After building the project with `npm run build`, the static files can be deployed to Firebase Hosting, making the web application available online with HTTPS and global CDN support. It requires minimal configuration and integrates smoothly with other Firebase services.


university-sports-reservation — esbuild ◂ npm run dev TERM_PROGRAM=Apple_Terminal SHELL=/bin/zsh TERM=xterm-256color —...

```
Last login: Mon Dec  1 05:22:38 on ttys000
(base) nurlanildirimli@Nurlans-Mac-mini university-sports-reservation % npm run dev

> university-sports-reservation@0.0.0 dev
> vite


  VITE v7.2.4  ready in 123 ms

  ➜  Local:   http://localhost:5173/
  ➜  Network: use --host to expose
  ➜  press h + enter to show help
```

11 GitHub Repository

Clone Command: git clone https://github.com/nurlanildirimli/University-Sports-Reservation.git

12 Milestones Remaining

Remaining Tasks

- UI polishing and responsive design
- User ban enforcement on frontend
- Optional analytics for admin
- Final testing across devices
- Performance optimization
- Writing final report

Timeline

- Nov 25–Dec 1 → UI polishing
- Dec 1–Dec 5 → Ban enforcement
- Dec 5–Dec 10 → Testing
- Dec 10–Dec 15 → Final report preparation

13 Final Deliverables

Source Code

- Frontend
- Firebase Functions

Cloud Functions Code

- Mail notification sending
- Banning the user who is marked as "Not Attended"

Documentation

- Proposal
- Progress Report
- Final Report (to be completed)

14 References

1. Firebase Documentation – Authentication, Firestore, Cloud Functions. `https://firebase.google.com/docs`

2. Google Cloud Storage – Object Storage Best Practices. `https://cloud.google.com/storage/docs/best-practices`