

- [Prerequisites](#)
- [Installation](#)
- [User Guide](#)
- [Current Features](#)
- [Features to be implemented](#)

Prerequisites

- [Python 3.13.2](#)
- [Node 22.14.0](#)
- [Firebase Project](#)

Installation

1. Clone the repository and navigate to the project directory:

```
git clone https://github.com/MdTanjeemHaider/ChronoMath
cd ChronoMath
```

2. Create a python virtual environment and activate it:

```
python -m venv venv

# on Windows
venv\Scripts\activate

# on Linux or MacOS
source venv/bin/activate
```

3. Install the required python packages:

```
pip install -r backend\chronomath\requirements.txt
```

4. Create a `.env` file in the `backend/chronomath/configs` directory and add the following lines:

```
FIREBASE_API_KEY=YOUR_FIREBASE_API_KEY
FIREBASE_AUTH_DOMAIN=YOUR_FIREBASE_AUTH_DOMAIN
FIREBASE_PROJECT_ID=YOUR_FIREBASE_PROJECT_ID
FIREBASE_STORAGE_BUCKET=YOUR_FIREBASE_STORAGE_BUCKET
FIREBASE_MESSAGING_SENDER_ID=YOUR_FIREBASE_MESSAGING_SENDER_ID
FIREBASE_APP_ID=YOUR_FIREBASE_APP_ID
FIREBASE_MEASUREMENT_ID=YOUR_FIREBASE_MEASUREMENT_ID
FLASK_SECRET_KEY=YOUR_FLASK_SECRET_KEY
```

5. Generate a new private key from the Firebase console and save it as `firebase_creds.json` in the `backend/chronomath/configs` directory.

6. Run the backend server:

```
python backend\chronomath\app.py
```

7. Open a new terminal and navigate to the project directory:

```
cd ChronoMath
```

8. Navigate to the frontend directory and install the required packages:

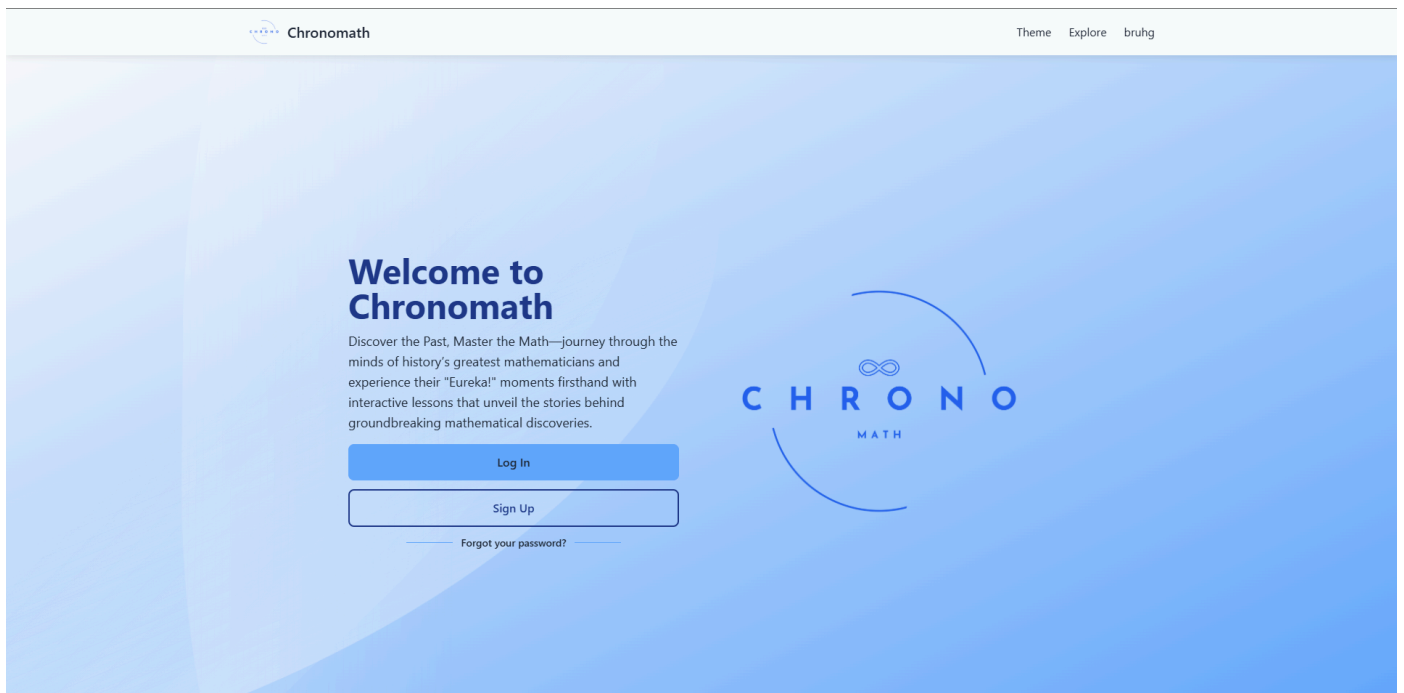
```
cd frontend\chronomath
npm install
```

9. Run the frontend server:

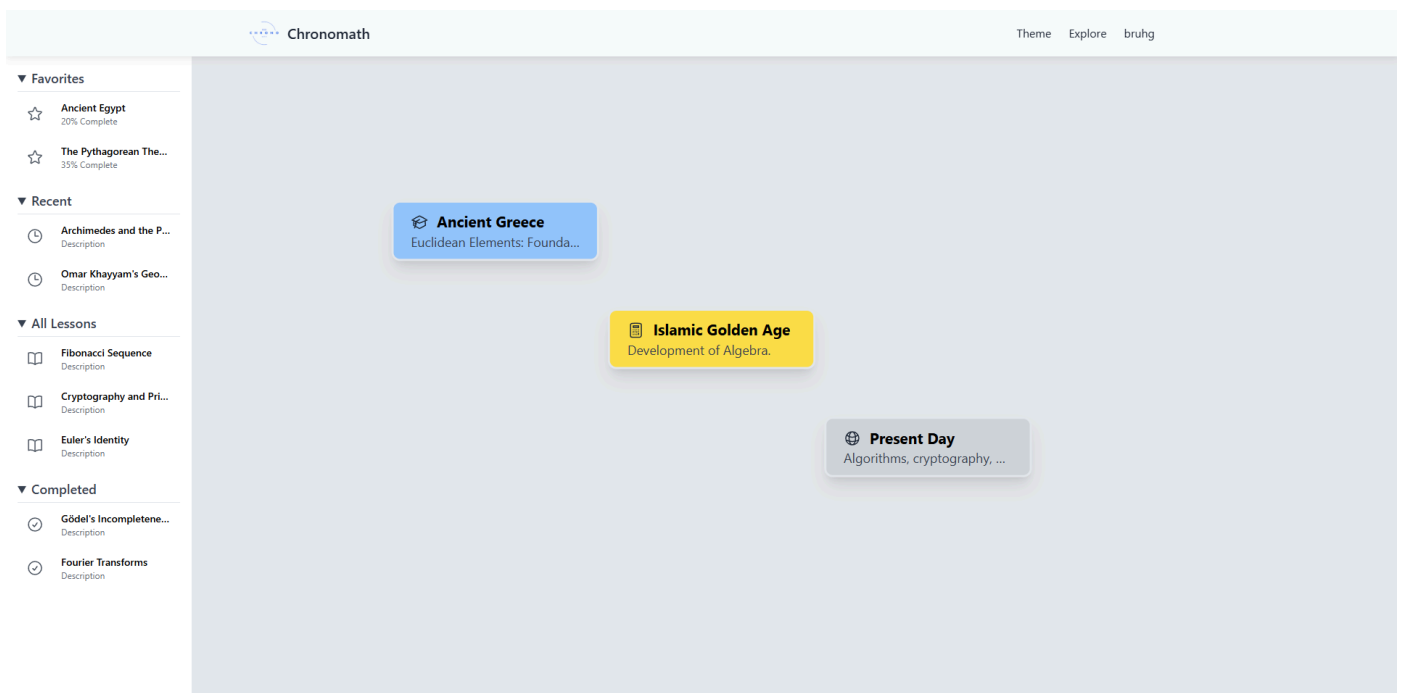
```
ng serve
```

User Guide

To begin, the user is greeted with the front page where they can either log in or sign up to create an account, either locally with an email or through Google integration



Once logged in, the user is greeted with the home page, showing the timeline structure of lessons, a sidebar to navigate favorite, recent, all, and completed lessons, and a top bar where they can choose a theme, go back to the home page, or check their profile.



When a user selects a course, a pop-up will appear giving a quick summary of the topics covered, and after continuing the user is greeted with various pages displaying the lessons, each categorized by the unit and any sub-sections in the unit.

Algebra Lessons

Quadratic Equations

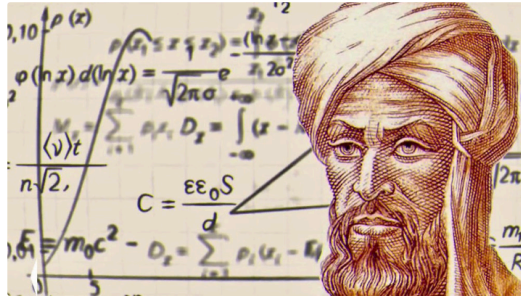
Lesson 1: Introduction

Lesson 2: Basic Elements & Forms

Lesson 3: Completion & Balancing

Lesson 4: Influence & Legacy

How Al-Khwarizmi's Work Became the Basis for Quadratic Equations & Legacy



Dissemination Through Islamic Lands and Latin Europe

The treatise spread across the Islamic world through copies and commentaries. Around the 12th century, it reached Europe via Latin translations (commonly attributed to translators such as Gerard of Cremona). These translations introduced the systematic treatment of equations into Western mathematics.

Influence on Later Mathematicians

Symbolic Notation:

While Al-Khwarizmi himself wrote in rhetorical style, his concepts of "roots," "squares," and "numbers" were instrumental in the future development of symbolic notation.

Standard Form:

The idea that any quadratic equation could be simplified to a standard form is still how we teach solving quadratics today (i.e., rewriting everything as $ax^2 + bx + c = 0$).

Current Features

Currently the major features implemented in the prototype include:

- Functional account creation when signing up and logging in implementation
- Home page with displayed timeline
- Lesson pages with unit, sub-section categorization with visuals implemented
- Quiz page
- Theme selection for users and accesbility themes for colorblind users
- Profile page

Features to be implemented

Major features that still need to be added and/or improved include:

- Overhaul of the timeline and visual upgrade
- Score Tracking
- Lesson page interactivity
- Notebook functionality
- Profile page settings functionality
- More learning objectives in general
- Favoriting system