SOFTWARE REQUIREMENTS SPECIFICATIONS

For

GURUQOOL

Team 6

202304002- Vanshil Soni

202304026-Mahir Shah

202304010-Dharmya Aagja

1. Introduction

Purpose

This document outlines the software requirements for **Guruqool**, an online one-on-one learning platform designed to connect learners with freelance tutors. The platform serves as a marketplace where users can learn skills across various domains, such as academics, music, art, and more, through personalized sessions.

Document Conventions

This SRS uses standard conventions: each requirement is uniquely numbered (e.g., REQ-1), and unless specified, higher-level requirement priorities are inherited by detailed requirements. Standard markdown formatting is used for clarity, with no special fonts or highlighting beyond this.

Intended Audience and Reading Suggestions

This document targets the following audiences:

- **Developers**: Focus on **Section 3 (External Interface Requirements)** and **Section 4 (System Features)** for implementation details.
- **Project Managers**: See **Section 2 (Overall Description)** for scope, constraints, and dependencies.
- **Testers**: Use **Section 4 (System Features)** to create test cases.
- **Documentation Writers**: Refer to **Section 1 (Introduction)** and **Section 2 (Overall Description)** for context, and **Section 5 (Other Nonfunctional Requirements)** for quality attributes.
- **Stakeholders**: Start with **Section 1** for an overview, then explore **Section 2** for product details.

The SRS is divided into five main sections, followed by appendices. Readers should begin with the **Introduction** and **Overall Description** for context before diving into role-specific sections.

Product Scope

Guruqool is an online marketplace facilitating personalized, one-on-one learning sessions. Key features include:

- User authentication and profile management for students and tutors.
- A searchable tutor marketplace with filters for domains, price, and availability.
- Real-time video call integration for sessions (to be implemented in a later phase).
- Order management for booking and tracking sessions.
- Transaction management for secure payments and withdrawals.
- An admin dashboard for user, order, and transaction oversight.

The platform aims to deliver a seamless, secure learning experience using modern web technologies.

References

This SRS aligns with general software engineering standards (e.g., IEEE SRS template) but does not reference specific external documents or websites at this stage.

---

2. Overall Description

Product Perspective

Guruqool is a standalone web application accessible via standard browsers. It integrates external APIs for video calls and payments to provide a comprehensive learning marketplace.

Product Functions

The platform offers the following core functions:

- User account creation, authentication, and profile management.
- A searchable marketplace with filters for tutors by domain, price, availability, and ratings.
- A booking system for scheduling one-on-one sessions.
- Real-time video call integration for sessions (phased implementation).
- Real-time chat for student-tutor communication.
- Order management for tracking booked and completed sessions.
- Transaction management for payments, balances, and withdrawals.

- An admin dashboard for managing users, approving tutors, and overseeing transactions.

User Classes and Characteristics

- **Admin**: Manages the platform, approves tutors, resolves issues, and monitors transactions. Requires full system access and administrative tools.
- **Student**: Books and attends sessions. Can search tutors, book sessions, join video calls, and provide feedback. Expected to have basic web usage skills.
- **Freelance Teacher (Tutor)**: Offers teaching services. Manages profile, availability, and sessions, and requests withdrawals. Must be skilled in their domain and have basic technical proficiency for web tools.

Operating Environment

The application will operate in:

- **Backend**: NodeJS
- **Frontend**: Angular or React
- **Database**: PostgreSQL
- **Browsers**: Chrome, Firefox, Safari, Edge

Design and Implementation Constraints

- Must scale to support multiple concurrent users, real-time chat (via websockets), and eventual video calls.
- Must securely handle user data and financial transactions.
- Mandated technologies: PostgreSQL (database), NodeJS (backend), Angular/React (frontend).

User Documentation

Provided documentation includes:

- **User Manuals**: Guides for students and tutors.
- **Online Help**: Embedded assistance in the app.
- **Admin Guide**: Instructions for management tasks.

Assumptions and Dependencies

Assumptions:

- Users have stable internet for video calls.
- Tutors ensure the quality of their teaching content.

Dependencies:

- External video call API.
- Payment gateway (e.g., Razorpay).
- Client-provided hosting server (e.g., Digital Ocean).
- PostgreSQL database system.

---

3. External Interface Requirements

User Interfaces

Designed in Figma and built with Angular/React, the UI features:

- Responsive design for desktop and mobile.
- Intuitive navigation for tutor search and booking.
- Real-time chat interface.
- Video call interface (later phase) with mute and screen-share controls.
- Admin dashboard with listings and management tools.

Hardware Interfaces

As a web app, **Guruqool** requires only standard devices (computers, tablets, smartphones) with browsers and internet access.

Software Interfaces

The app integrates with:

- **PostgreSQL Database**: Stores user data, sessions, and transactions.
- **NodeJS Backend**: Manages logic and API requests.
- **Angular/React Frontend**: Handles user interaction.
- **External APIs**: Video call and payment processing (e.g., Razorpay).

Communications Interfaces

The app uses:

- **HTTPS**: Secure data transmission.
- **Websockets**: Real-time chat.
- **Video Call API**: Session integration (later phase).

● **Database Queries**: PostgreSQL data management.

---

4. System Features

System Feature 1: Search

Description and Priority

The search feature lets students find tutors by domain, price, availability, and ratings. **High priority** due to its core role in usability.

Stimulus/Response Sequences

- **Stimulus**: Student enters keywords or applies filters.
- **Response**: System lists matching tutors (name, domain, price, rating).
- **Stimulus**: Student sorts by price or rating.
- **Response**: System updates the list.
- **Stimulus**: Student selects a tutor.
- **Response**: System shows the tutor's profile and booking options.

Functional Requirements

- **REQ-1**: Provide a search bar for keywords.
- **REQ-2**: Enable filters for domain, price, availability, and ratings.
- **REQ-3**: Display results with tutor details (name, domain, price, rating).
- **REQ-4**: Allow sorting by price, rating, and availability.
- **REQ-5**: Shows results exceeding 10 entries per page.

System Feature 2: Video Call

Description and Priority

The video call feature supports one-on-one sessions. **Low priority** due to implementation complexity; chat will be prioritized, with video calls added later.

Stimulus/Response Sequences

- **Stimulus**: Student books a session.
- **Response**: System schedules and notifies both parties.
- **Stimulus**: Session time arrives; both join the call.
- **Response**: System launches the video interface with controls.
- **Stimulus**: Session ends.

- **Response**: System marks it completed and updates order status.

Functional Requirements

- **REQ-6**: Integrate a video call API.
- **REQ-7**: Provide mute, video disable, and screen-share controls.
- **REQ-8**: Notify users of session start times.
- **REQ-9**: Mark sessions as completed upon ending.

System Feature 3: Order Management

Description and Priority

Tracks orders and allows completion marking. **Medium priority** for tracking and admin purposes.

Stimulus/Response Sequences

- **Stimulus**: Student books a session.
- **Response**: System adds it to the order list.
- **Stimulus**: Tutor completes the session.
- **Response**: System allows marking as completed.
- **Stimulus**: Admin views orders.
- **Response**: System displays a detailed, filterable list.

Functional Requirements

- **REQ-10**: Maintain a list of running orders.
- **REQ-11**: Allow tutors to mark orders completed.
- **REQ-12**: Provide admins with a detailed order list.

System Feature 4: Transaction Management

Description and Priority

Manages transactions, balances, and withdrawals. **High priority** due to financial importance.

Stimulus/Response Sequences

- **Stimulus**: Student pays for a session.
- **Response**: System records the transaction and updates balances.
- **Stimulus**: Tutor requests a withdrawal.
- **Response**: System lists it for admin approval.

- **Stimulus**: Admin approves it.
- **Response**: System processes the withdrawal and updates balances.

Functional Requirements

- **REQ-13**: List transactions with details (amount, date, status).
- **REQ-14**: Track balances and allow withdrawal requests.
- **REQ-15**: Enable admins to approve/reject withdrawals.

---

5. Other Nonfunctional Requirements

Performance Requirements

- Handle 10 concurrent video sessions without issues.
- Page load times under 2 seconds.
- Search results displayed within 1 second.

Safety Requirements

- Use Razorpay for secure financial transactions.
- Encrypt user data during transmission and storage.

Security Requirements

- Implement authentication with password hashing.
- Enforce role-based access (admin, student, tutor).
- Use parameterized queries to prevent SQL injection.
- Sanitize inputs to avoid XSS attacks.

Software Quality Attributes

- **Scalability**: Support a growing user base.
- **Reliability**: Achieve 99.9% uptime.
- **Usability**: Offer an intuitive interface with minimal training needed.

Business Rules

- Tutors require admin approval to offer services.
- Students can only book available tutors.
- Payments are held in escrow until session completion.

Other Requirements

- Support future internationalization if needed.
- Comply with data protection laws.

Appendix A: Glossary

- **Tutor**: A freelance teacher offering sessions.
- **Order**: A booked session.
- **Transaction**: A financial exchange (payment or withdrawal).
- **Admin**: A user role with full access to manage the platform, including approving tutors, handling user issues, and overseeing transactions.
- **Authentication**: The process of verifying a user's identity, typically through credentials like username and password or multi-factor methods.
- **Availability Calendar**: A tutor-specific feature showing their open time slots for booking sessions.
- **Booking System**: The platform mechanism allowing students to schedule sessions with tutors based on availability and preferences.
- **Dashboard**: A personalized interface for users (students, tutors, or admins) to view key information like upcoming sessions, earnings, or system stats.
- **Escrow**: A financial arrangement where a third party holds payments until the session is completed, ensuring secure transactions.
- **Feedback**: Reviews or ratings provided by students about tutors (or vice versa) after a session, influencing tutor credibility.
- **Filter**: A search feature that allows students to narrow down tutor results based on criteria like domain, price, or availability.

Wireframe

[Homepage]

Explore

Redefine Boundries

Body text for whatever you'd like to expand on the main point.

Convinience of selecting mutual time

Body text for whatever you'd like to add more to the main point. It provides details, explanations, and context.

Know More   Register as a Guru

Languages

Learn new languages

Handicrafts

The Handicrafts of old Ahmedabad!

Languages
Learn new languages

Handicrafts
The Handicrafts of old Ahmedabad!

Our Reviews

"A terrific piece of praise"

Name
Description

"A fantastic bit of feedback"

Name
Description

"A genuinely glowing review"

Name
Description

About us

Help   Up

Guruqool

Terms and Condtions

Cookie Policy

Cookie settings

Find out more

Help

How it works

Tutor Area

Join us

Become a tutor

Students looking for tutor

News and Blog

[Search Page]

Music tutors and lessons

🔔 Save this search

What would you like to learn?

Music ✕

Where?

E.g.: London, Leeds...

Lesson location
☐ In person
☐ Online

Advanced features

Advanced features...

Bethany
• Online
⊚ Southampton, Marchwood, Ports...
☑ Music

£30/hr
First class free

Music Graduate teaching music theory, piano and all music related subjects

I am a recent music graduate. Spending the last 3 years studying my Masters in Switzerland - specialising in Renaissance music. I also stud...

Highlighted

Simon James Collier
⊚ Brunswick Village, Ponteland,...
☑ Music

£26/hr
First class free
4.5 (2 Reviews)

Private Tuition in Violin&Viola & Elementary Piano

I have a Private Teacher's Licentiate Diploma (LTCL) in Viola. I currently do a little Private Teaching, I give 30 mins lessons for beginne...

Highlighted

Malcolm
• Online
⊚ Bramley (Surrey), Artington, ...
☑ Music

£30/hr
First class free

Willing to travel, friendly prices, surrey abd surroundings , piano, violin and viola

Im a professional musician, performing on the violin and teaching for over 30 years, absolutely love to help people who want to learn, i ca...

Kevin
⊚ Derry, Drumahoe, Foyle Hill, ...
☑ Music

£25/hr
First class free

Guitar teacher available to teach all levels,ages,genres and topics

I am a music teacher based in Derry/Londonderry with several years experience playing in bands and teaching guitar and music both in school...

Still looking for a tutor?
Post an advert and tutors will be able to contact you

Post your ad

Guruqool

Terms and Condtions	Find out more	Join us
Cookie Policy	Help	Become a tutor
Cookie settings	How it works	students looking for tutor
	Tutor Area	News and Blog

🅕 in ▶ 🅞

12

[Guru Profile Page]

Guru   Post your ad   Help   My account

About Guru

Subheading for description or instructions

Body text for your whole article or post. We'll put in some lorem ipsum to show how a filled-out page might look:

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed to using 'Content here, content here', making it look like readable English. Many desktop publishing packages and web page editors now use Lorem Ipsum as their default model text, and a search for 'lorem ipsum' will uncover many web sites still in their infancy. Various versions have evolved over the years, sometimes by accident, sometimes on purpose (injected humour and the like).

Contact Guru

First name

Jane

Last name

Smitherton

Email address

email@janesfakedomain.net

Your message

Enter your question or message

Submit

Guruqool

Terms and Condtions

Cookie Policy

Cookie settings

Find out more

Help

How it works

Tutor Area

Join us

Become a tutor

Students looking for tutor

News and Blog