

SRS Template

PROJECT NAME: [Your App Name]

VERSION: 1.0

DATE: [Today's Date]

1. PROJECT OVERVIEW

One paragraph describing what your app does and why it exists

Example: "BookFlow is a fitness class booking platform that helps yoga instructors manage schedules and collect payments without paying high commission fees to platforms like ClassPass."

2. BUSINESS OBJECTIVES

- What problem does this solve?
- What business model are you using? (free, paid, commission, freemium?)
- What's your target market size?

Example:

- Solve: Yoga instructors spending 5 hours weekly on schedule management
- Model: \$50-\$100/month SaaS subscription per instructor
- Market: 15,000+ independent yoga studios in the US

3. USER PERSONAS (2-3 maximum)

For each user type, document:

- Who are they? (title, industry, age, background)
- What problem do they have?
- What's their technical comfort level?
- What outcomes do they want?

Example Persona 1:

- Name: Sarah
- Role: Full-time yoga instructor, independent studio
- Problem: Manages 15 classes weekly, tracks bookings via email
- Tech comfort: Moderate (comfortable with apps, not technical)
- Goal: Spend <30 min/week on administration

Example Persona 2:

- Name: Michael
- Role: Studio owner with 5 instructors
- Problem: No unified system across multiple instructors' schedules
- Tech comfort: Low (prefers simple solutions)
- Goal: Single dashboard to see all classes and revenue

4. CORE FEATURES (Top 5-7, no more)**List ONLY features critical for MVP**

For each feature, explain: What? Who uses it? Why do they need it?

Example:**Feature 1: Class Management**

- What: Instructors create, edit, and publish classes
- Who: Instructors
- Why: Instructors need a way to list their available classes

Feature 2: Student Booking

- What: Students browse available classes and reserve spots
- Who: Students
- Why: Students need to see when classes are available and book

Feature 3: Payment Processing

- What: Accept payments, process transactions, pay out instructors
- Who: Students (pay), System (processes), Instructors (receive)
- Why: Revenue model—instructors keep 95%, platform takes 5%

Feature 4: Booking Confirmations

- What: Automated emails confirming bookings
- Who: Students receive, Instructors see in dashboard
- Why: Reduces no-shows and provides transaction records

Feature 5: Attendance Tracking

- What: Instructors mark who attended, export attendance records
- Who: Instructors
- Why: Instructors need records for business reporting

5. USER WORKFLOWS (Step-by-step flows for core actions)

WORKFLOW 1: How an Instructor Lists a Class

1. Instructor logs in
2. Clicks "Create Class"
3. Enters: Class name, description, date, start time, end time, capacity, price
4. Selects: Difficulty level, class type (yoga, pilates, etc.)
5. Click "Publish"
6. Class appears in public booking interface
7. Instructor sees confirmation: "Class published successfully"

WORKFLOW 2: How a Student Books a Class

1. Student discovers app (website, app store, friend referral)
2. Click "Browse Classes"
3. Sees all available classes (filtered by date, location, type)
4. Clicks on specific class
5. Sees: Class details, instructor bio, reviews, available spots, price
6. Click "Book Now"
7. Payment screen appears (Stripe)
8. Enters payment info
9. Payment processes
10. Student sees confirmation + email arrives with booking details
11. Calendar event added to student's calendar (optional)

WORKFLOW 3: How Instructor Manages Revenue

1. Instructor logs into dashboard
2. Sees: Revenue this month, upcoming classes, recent bookings
3. Clicks "Payments"
4. Sees: Monthly revenue breakdown, commission taken by platform, payout amount
5. Clicks "Withdraw Funds"
6. Funds transfer to instructor's bank account (automatic weekly payouts)

6. DATA REQUIREMENTS

What information needs to be stored? How does it relate?

USERS TABLE

- User ID (unique identifier)
- Email
- Password (encrypted)
- Name
- User Type (Instructor or Student)
- Phone number
- Address (optional)
- Created date

CLASSES TABLE

- Class ID
- Instructor ID (who teaches this class)
- Class name
- Description
- Date
- Start time
- End time
- Capacity (max students)
- Current bookings (how many registered)
- Price
- Class type (yoga, pilates, etc.)
- Status (published, draft, cancelled)

BOOKINGS TABLE

- Booking ID
- Student ID (who booked)
- Class ID (which class)
- Booking date
- Payment status (completed, pending, failed)
- Attendance (attended, no-show, cancelled)
- Amount paid

PAYMENTS TABLE

- Payment ID
- Booking ID
- Amount

- Platform commission (5%)
- Instructor payout (95%)
- Payment date
- Payout date (when money reaches instructor account)
- Status (completed, pending, failed, refunded)

7. TECHNICAL INTEGRATIONS

What services does the app need to connect to?

- Payment Processing: Stripe (accept card payments)
- Email Service: SendGrid or Mailgun (send confirmations, reminders)
- Calendar: Google Calendar API (optional - sync classes to student calendars)
- Analytics: Amplitude or Mixpanel (track user behavior)
- Hosting: [Your chosen platform's hosting]
- Database: [Your chosen platform's database]

8. MUST-HAVE vs NICE-TO-HAVE

MUST-HAVE (Launch with these):

- Instructor sign-up and login
- Instructor creates and publishes classes
- Student browse and book classes
- Payment processing (Stripe)
- Booking confirmation emails
- Basic dashboard (instructors see bookings)
- Attendance tracking

NICE-TO-HAVE (Add later):

- Instructor reviews and ratings
- Class cancellation with automatic refunds
- Waiting lists for full classes
- Calendar sync (Google Calendar)
- SMS reminders

- Class pass/package pricing
- Student scheduling preferences
- Instructor scheduling templates

9. SUCCESS METRICS & GOALS (How will you measure success?)

Month 1-3 Goals (Validation phase):

- 50-100 instructor sign-ups
- 500+ bookings
- 20%+ repeat booking rate
- \$1,000+ monthly revenue
- <2% payment failure rate

Month 3-6 Goals (Growth phase):

- 500+ instructors
- 5,000+ monthly bookings
- 30%+ repeat rate
- \$10,000+ monthly revenue
- 90%+ payment success rate

Ongoing Metrics:

- Active instructors (month-to-month)
- Bookings per instructor (growth indicator)
- Student retention rate (% who book multiple times)
- Revenue per instructor (LTV)
- System uptime (target: 99.5%)
- Customer satisfaction (NPS score)

10. CONSTRAINTS & ASSUMPTIONS

Constraints:

- Budget: Under \$1,000 for MVP development
- Timeline: Launch within 6 weeks
- Team: Solo founder, non-technical background
- Initial platform: [Your chosen no-code platform name]

Assumptions:

- Target instructors have email and basic smartphone
- Instructors willing to pay \$50-100/month
- Students willing to use app for booking
- Payment processing smooth with Stripe
- Can launch MVP with 5 core features

- Will have real users within 4 weeks

11. OUT OF SCOPE (What you're NOT building)

For MVP, we are NOT building:

- Mobile app (web-only for now, responsive design)
- Advanced scheduling algorithms
- AI-powered class recommendations
- Instructor certification system
- Community features (forums, messaging)
- Video classes/live streaming
- Equipment rental integration
- Integration with gym management systems

These may be added in future versions if market validates need.