

Contents

Product Requirements Document (PRD)	1
Wedding Invitation & RSVP Automation App	1
1. Overview	1
2. Users and Roles	2
3. Scope	3
4. Core Concepts and Data Model	3
5. Functional Requirements	6
6. User Journeys	13
7. Technical Architecture	14
8. Integrations	17
9. Design and Content Requirements	18
10. Delivery Plan	19
11. Testing Requirements	20
12. Risks and Mitigations	20
13. Open Questions	22
14. Appendix	22

Product Requirements Document (PRD)

Wedding Invitation & RSVP Automation App

Field	Value
Document version	1.0
Date	2 August 2026
Owner	Mod Murad
Status	Draft for review
Target stack	Next.js (App Router) + PostgreSQL + Prisma
Channels	Email, WhatsApp, SMS

1. Overview

1.1 What we are building (in plain language)

A small web application that replaces the paper-and-phone-call way of managing wedding guests.

The host loads a guest list into the system. Each guest receives a **personal link** by WhatsApp, SMS, or email. When the guest taps the link, a beautiful animated invitation card opens in their browser — like a digital card that unfolds itself. At the bottom there is an **“Accept Invitation”** button. Only after they tap Accept does a short form appear asking for their **name, email and phone number**. Once submitted, they are counted as attending.

There are **three separate events** — Mehedi, Marriage Ceremony, and Walima — and each has its own invitation card and its own headcount. A guest can be invited to one, two, or all three.

Printed cards will also be handed out. Each printed card carries a **QR code**. Scanning it opens the same invitation page.

Behind the scenes, the app runs an **automatic reminder engine**. Every guest who accepted gets a reminder **15 days, 7 days, and 2 days** before their event. Each reminder contains a **cancel link**, so a guest whose plans change can withdraw with one tap — and the headcount updates instantly.

The host logs into an **admin dashboard** to see live numbers: invited, accepted, declined, pending, cancelled — per event and in total. They can export the full guest list with contact details as a **CSV file** for the caterer or venue.

1.2 Why it matters

Problem today	How this app fixes it
Headcount is guessed from phone calls and memory	Live, per-event, always-accurate count
Catering over- or under-ordered; money wasted	Export exact numbers 2 days before each event
No record of who was invited to which event	One database, three event lists
Manual reminder calls to hundreds of people	Automated 15/7/2-day reminders on 3 channels
Last-minute dropouts discovered on the day	Guests self-cancel via a link; count auto-adjusts
Printed cards give no feedback	QR code brings paper guests into the same system

1.3 Success criteria

Metric	Target
Invitation delivery rate (all channels)	≥ 95%
Guests who open their link	≥ 70%
Guests who respond (accept or decline)	≥ 60% within 7 days
Final headcount accuracy vs. actual attendance	Within ±10%
Invitation page load on 3G mobile	< 2.5 seconds
Admin can export CSV	< 3 clicks, any time
Reminder delivered on the correct day	100% (no missed schedules)

2. Users and Roles

2.1 Personas

Host / Admin (the couple or a family member) Not technical. Manages the guest list, sends invitations, watches the numbers, downloads the CSV. Uses a phone as much as a laptop. Needs the dashboard to be obvious without training.

Co-host (optional, e.g. bride's brother) Same as admin but should not be able to delete the whole guest list or change gateway settings.

Invited Guest (tokenized) Received a personal link on WhatsApp/SMS/email. Ages 18–75, mostly on mobile, mixed technical comfort. Wants to see a beautiful card, tap one button, and be done in under 60 seconds. Many will read Bangla more comfortably than English.

Walk-in Guest (QR / open link) Got a printed card, scanned the QR. The system does not yet know who they are, so they must type their details themselves.

2.2 Role permissions

Capability	Super Admin	Co-host	Viewer
View dashboard & counts	Yes	Yes	Yes
Add / edit / import guests	Yes	Yes	No
Delete guests	Yes	No	No
Send invitations & broadcasts	Yes	Yes	No
Export CSV	Yes	Yes	Yes
Edit invitation content / animation	Yes	Yes	No
Configure gateways & API keys	Yes	No	No
Manage admin users	Yes	No	No
View audit log	Yes	No	No

3. Scope

3.1 In scope — v1 (MVP)

1. Three event types with independent invitation pages: **Mehedi, Marriage Ceremony, Walima**.
2. Animated, mobile-first invitation landing pages.
3. “Accept Invitation” flow that reveals the RSVP form only after acceptance.
4. Guest identification via **unique token links** and an **open QR / public link**.
5. Admin dashboard with per-event and aggregate counts.
6. Guest list management: manual add, bulk CSV import, edit, tag by event.
7. CSV export of guests with name, email, phone, event, status, timestamps.
8. Outbound messaging via **Email + WhatsApp + SMS**.
9. Automated reminders at **T-15, T-7, T-2 days** per event.
10. Self-service **cancellation** link in every reminder.
11. QR code generation for printed cards (one open QR + optional per-guest QR).
12. Delivery status tracking (sent / delivered / failed / read where available).
13. Bangla + English content support on the invitation page.
14. Basic audit log of admin actions.

3.2 Out of scope — v1 (candidates for v2)

- Guest-to-guest features (seating charts, table assignments, photo uploads)
- Gift registry or online payments
- Multi-tenant SaaS (multiple unrelated weddings on one instance)
- Live-streaming integration
- Native mobile apps
- AI-generated card designs
- Plus-one management with named sub-guests (v1 only stores a plus-one *count*)

3.3 Assumptions

- Single wedding, single host family (multi-tenancy not needed in v1).
 - Expected scale: **300–1,500 guests**, up to ~3,000 messages per reminder wave.
 - Guests have smartphones with a browser; no app install required.
 - Host is willing to pay for WhatsApp Business API template messages (~BDT 0.7–1.5 per message) and an SMS gateway.
 - Wedding dates are fixed at least 30 days before the first event, so all three reminder waves can fire.
-

4. Core Concepts and Data Model

4.1 The mental model

Wedding (1)

└ Event (3: Mehedi, Marriage, Walima)

└ Invitation (one per guest per event) ← carries the unique token

└ RSVP (the guest's answer + contact details)

└ Reminder Jobs (T-15, T-7, T-2)

└ Message Log (one row per send attempt)

Guest (1 person, may hold 1–3 Invitations)

The key idea: **the Invitation is the unit of tracking, not the Guest**. A guest invited to all three events has three invitations and can accept two and decline one.

4.2 Entities

wedding

Field	Type	Notes
id	uuid PK	
bride_name, groom_name	text	Shown on cards
slug	text unique	e.g. murad-weds-x
default_locale	enum(bn,en)	
timezone	text	Default Asia/Dhaka
created_at	timestampz	

event

Field	Type	Notes
id	uuid PK	
wedding_id	uuid FK	
type	enum(mehedi,marriage,walima)	
title_bn, title_en	text	
starts_at	timestampz	Drives all reminder math
ends_at	timestampz	nullable
venue_name, venue_address	text	
map_url	text	Google Maps link
dress_code, notes	text	nullable
theme_key	text	Which animated template to use
cover_image_url, music_url	text	nullable
rsvp_deadline	timestampz	After this, page shows "RSVP closed"
capacity	int	nullable; used for over-capacity warnings
is_published	boolean	Draft events are not publicly reachable

guest

Field	Type	Notes
id	uuid PK	
wedding_id	uuid FK	
full_name	text	
email	text	nullable, validated
phone_e164	text	nullable, normalised to +8801XXXXXXXXXX
whatsapp_phone_e164	text	nullable, defaults to phone
side	enum(bride,groom,common)	For filtering
group_tag	text[]	e.g. family, office, university
preferred_locale	enum(bn,en)	
preferred_channel	enum(whatsapp,sms,email,auto)	
source	enum(manual,csv_import,self_registered)	
is_deleted	boolean	Soft delete
created_at, updated_at	timestampz	

Uniqueness rule: (wedding_id, phone_e164) and (wedding_id, lower(email)) are unique where not null. This prevents duplicate guests on CSV re-import and on QR self-registration.

invitation

Field	Type	Notes
id	uuid PK	
event_id	uuid FK	

Field	Type	Notes
guest_id	uuid FK	nullable — null for
token	text unique	22-char URL-safe
short_code	text unique	6-char human-type
status	enum(pending,opened,accepted,declined,cancelled,expired)	
max_guests	int default 1	Ceiling for plus-c
opened_at, responded_at, cancelled_at	timestampz	nullable
open_count	int default 0	
invited_via	enum(whatsapp,sms,email,print_qr,link)	
created_at, updated_at	timestampz	

rsvp

Field	Type	Notes
id	uuid PK	
invitation_id	uuid FK unique	One current answer per invitation
response	enum(accepted,declined,cancelled)	
name, email, phone_e164	text	As typed by the guest on the form
party_size	int default 1	1 = just them
adults, children	int	nullable breakdown
message_to_couple	text	nullable, max 500 chars
dietary_notes	text	nullable
responded_at	timestampz	
ip_hash, user_agent	text	Abuse detection only

rsvp_history Append-only log of every status change (pending → accepted → cancelled) with timestamp, actor (guest/admin), and source. Needed because a cancellation must not erase the fact that they once accepted.

message_template

Field	Type	Notes
id	uuid PK	
channel	enum(email,whatsapp,sms)	
purpose	enum(invite,reminder_15,reminder_7,reminder_2,thank_you,cancel_confirm,c)	
locale	enum(bn,en)	
subject	text	Email only
body	text	With {{placeholders}}
provider_template_id	text	WhatsApp approved template name
is_active	boolean	

reminder_schedule

Field	Type	Notes
id	uuid PK	
event_id	uuid FK	
offset_days	int	15, 7, 2 (configurable)
send_at_local_time	time	e.g. 10:00 Asia/Dhaka
channels	text[]	e.g. {whatsapp,email}
audience	enum(accepted,pending,all)	

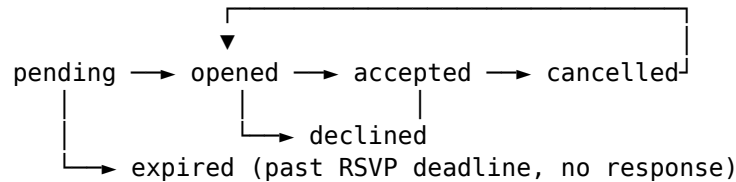
Field	Type	Notes
is_enabled	boolean	

message_job One row per (invitation × schedule × channel). This is the queue. | Field | Type | Notes | |---| | id | uuid PK | | invitation_id, template_id, schedule_id | uuid FK | | channel | enum | | scheduled_for | timestampz | Computed absolute send time | | status | enum(queued,sending,sent,delivered,read,failed,skipped) | | attempts | int | | provider_message_id | text | For webhook matching | | error_code, error_message | text | | idempotency_key | text unique | invitation:schedule:channel — prevents double sends | | sent_at, delivered_at | timestampz |

admin_user id, email, password_hash (argon2id), role, name, last_login_at, is_active, totp_secret (optional 2FA).

audit_log id, admin_user_id, action, entity_type, entity_id, before/after JSON, ip, created_at.

4.3 Status lifecycle



Rules: - accepted → cancelled is allowed any time before the event starts. - cancelled → accepted is allowed (guest changes mind again) — the same link works. - declined → accepted is allowed. - Every transition writes an rsvp_history row. - Headcount only counts invitations currently in accepted state, summing party_size.

5. Functional Requirements

Priority key: **P0** = must have for launch, **P1** = should have, **P2** = nice to have.

5.1 Guest-facing invitation page

ID	Requirement	Priority
FR-1.1	Route /i/{token} loads the invitation for that specific guest and event, pre-filling their known name.	P0
FR-1.2	Route /e/{eventSlug} is the open/QR link — no token, guest is unknown until they register.	P0
FR-1.3	Page renders a full-screen animated card: envelope-open or curtain-reveal intro, then scrolling sections.	P0
FR-1.4	Sections: Hero (names + date), Countdown timer, Event details (time, venue, map), Story/photo gallery (optional), Accept/Decline call-to-action, Contact & directions footer.	P0

ID	Requirement	Priority
FR-1.5	Guest's name is personalised in the hero ("Dear Rahim Uddin, you are invited...") when a token is present.	P0
FR-1.6	Optional background music with a clearly visible mute toggle; must never autoplay with sound (browsers block it and it annoys people). Default = muted, tap to unmute.	P0
FR-1.7	Language toggle Bangla <-> English; default from preferred_locale.	P1
FR-1.8	"Add to Calendar" button generating a .ics file and Google Calendar link.	P1
FR-1.9	"Get Directions" button opening Google Maps.	P0
FR-1.10	If the guest is invited to more than one event, show a discreet switcher: "You're also invited to Walima →".	P1
FR-1.11	If rsvp_deadline has passed, hide the form and show a polite closed message with a contact number.	P0
FR-1.12	If the invitation is already accepted, show their current answer with an "Update my response" option instead of a blank form.	P0
FR-1.13	Page must be fully usable one-handed on a 360px-wide screen.	P0
FR-1.14	Share button (Web Share API) so guests can forward the open link to family.	P2

5.2 Accept flow and RSVP form

ID	Requirement	Priority
FR-2.1	The RSVP form is hidden until the guest taps "Accept Invitation" — then it slides/fades in and the page scrolls to it.	P0
FR-2.2	Form fields: Name (required), Phone (required), Email (optional but encouraged) , Number of guests attending (1..max_guests), Message to the couple (optional), Dietary notes (optional).	P0

ID	Requirement	Priority
FR-2.3	Phone input with country selector defaulting to BD +880; auto-normalise 01712345678 → +8801712345678; reject invalid lengths inline.	P0
FR-2.4	Email validated by format; if provided, used for the email reminder channel.	P0
FR-2.5	Known fields are pre-filled from the guest record; guest may correct them, and corrections update the guest record.	P0
FR-2.6	A separate “Sorry, I can’t make it” button records declined and shows a short “we’ll miss you” screen — no form required.	P0
FR-2.7	On successful submit: confetti/thank-you animation, summary of what they confirmed, Add-to-Calendar, and a note that they can cancel later using the same link.	P0
FR-2.8	An immediate confirmation message is queued on the guest’s preferred channel (and email if given).	P0
FR-2.9	Submitting twice must not create two RSVPs (idempotent on <code>invitation_id</code>).	P0
FR-2.10	On the open/QR link , the form additionally asks which event(s) they are responding to (checkboxes), then creates the guest + invitation records.	P0
FR-2.11	Open-link submissions with a phone/email that already exists are merged into the existing guest, not duplicated.	P0
FR-2.12	Rate limit: max 5 submissions per IP per 10 minutes; hCaptcha/Turnstile shown on the open link only.	P0
FR-2.13	Form works without JavaScript as a plain POST fallback (progressive enhancement).	P2

5.3 Cancellation flow

ID	Requirement	Priority
FR-3.1	Every reminder and confirmation message contains a cancel link: <code>/i/{token}/cancel</code> .	P0

ID	Requirement	Priority
FR-3.2	The cancel page shows what they are cancelling (event, date, party size) and requires one confirmation tap — no accidental cancels from link-preview crawlers.	P0
FR-3.3	Cancellation must be a POST , never a bare GET, so WhatsApp/email link-preview bots cannot trigger it.	P0
FR-3.4	Optional one-line reason field.	P2
FR-3.5	On cancel: status → cancelled, headcount decrements immediately, all future reminder jobs for that invitation are marked skipped.	P0
FR-3.6	Guest sees a “changed your mind?” re-accept button; admin sees a cancellation notification on the dashboard.	P1
FR-3.7	Cancellation is blocked after the event start time (shows contact number instead).	P1

5.4 Admin dashboard

ID	Requirement	Priority
FR-4.1	Login at /admin with email + password; session cookie, httpOnly, 12-hour expiry. Optional TOTP 2FA.	P0
FR-4.2	Overview screen: three event cards, each showing Invited / Accepted (with total headcount incl. plus-ones) / Declined / Pending / Cancelled, plus a response-rate progress bar and days-to-event countdown.	P0
FR-4.3	Aggregate row: unique guests, total accepted headcount across events, overall response rate.	P0
FR-4.4	Response trend chart (accepts per day) and channel breakdown (which channel produced responses).	P1
FR-4.5	Guest table with search, and filters by event, status, side (bride/groom), group tag, channel, date range.	P0
FR-4.6	Inline actions per guest: view detail, edit, resend invitation, mark accepted/declined manually (for phone RSVPs), delete.	P0

ID	Requirement	Priority
FR-4.7	Guest detail drawer showing full timeline: invited → opened → accepted → reminders sent → cancelled, with timestamps and channels.	P1
FR-4.8	Add guest form and Bulk CSV import with column mapping, a preview of the first 10 rows, duplicate detection, and a per-row error report.	P0
FR-4.9	Send screen: choose event → choose audience (all / pending only / accepted only / by tag) → choose channel(s) → preview rendered message → send now or schedule. Shows an estimated cost and message count before sending.	P0
FR-4.10	Message log with per-message status, provider ID, error reason, and a “retry failed” bulk action.	P0
FR-4.11	Reminder settings per event: enable/disable each of T-15, T-7, T-2, choose send time and channels, preview upcoming waves with exact dates and recipient counts.	P0
FR-4.12	Content editor per event: titles, date/time, venue, map link, cover image, gallery images, theme selection, and Bangla/English text — with a live phone-frame preview.	P1
FR-4.13	QR code generator : download the event's open-link QR as PNG and print-ready SVG/PDF at 300 DPI; optionally generate a per-guest QR sheet for personalised printed cards.	P0
FR-4.14	Settings : gateway credentials, sender IDs, wedding details, admin users, rate limits.	P0
FR-4.15	Dashboard counts refresh live (poll every 30s or SSE).	P1
FR-4.16	Dashboard is usable on a phone.	P1
FR-4.17	Capacity warning banner when accepted headcount exceeds the event capacity.	P2

5.5 CSV export

ID	Requirement	Priority
FR-5.1	“Download CSV” button on the dashboard and on the guest table, respecting the currently applied filters.	P0
FR-5.2	Columns: guest_id, full_name, email, phone, whatsapp, side, group_tags, event, status, party_size, adults, children, dietary_notes, message_to_couple, invited_via, invited_at, opened_at, responded_at, cancelled_at, last_reminder_sent_at, notes.	P0
FR-5.3	UTF-8 with BOM so Bangla names render correctly when opened in Microsoft Excel.	P0
FR-5.4	Filename: guests_{event}_{status}_{YYYY-MM-DD_HHmm}.csv.	P0
FR-5.5	Separate one-click exports: “Caterer list” (accepted only: name, party size, dietary notes) and “Full list”.	P1
FR-5.6	Exports > 5,000 rows stream rather than buffering in memory.	P1
FR-5.7	Every export writes an audit log entry (who exported what, when) – this is personal data leaving the system.	P0
FR-5.8	XLSX export option.	P2

5.6 Messaging and automation

ID	Requirement	Priority
FR-6.1	Support three channels: Email (transactional provider), WhatsApp (Business Cloud API), SMS (local Bangladesh gateway).	P0
FR-6.2	Channel selection strategy per guest: use preferred_channel; if auto, try WhatsApp → fall back to SMS on failure → fall back to Email. Fallback attempts are capped at 2 per message.	P0

ID	Requirement	Priority
FR-6.3	Templates support placeholders: {{guest_name}}, {{event_title}}, {{event_date}}, {{event_time}}, {{venue}}, {{days_left}}, {{invite_url}}, {{cancel_url}}, {{map_url}}, {{couple_names}}.	P0
FR-6.4	Every outbound message carries a short link (/i/{token}) – links must be under 60 characters so SMS stays within 1–2 segments.	P0
FR-6.5	Reminders fire at T-15, T-7 and T-2 days before event.starts_at, at a configurable local time (default 10:00 Asia/Dhaka).	P0
FR-6.6	Reminder audience default = accepted. Optionally a parallel “still waiting for your reply” nudge to pending guests.	P0
FR-6.7	If the app is set up fewer than 15 (or 7) days before an event, past waves are skipped automatically, not fired late.	P0
FR-6.8	Idempotency: the same guest can never receive the same reminder twice, even if the scheduler runs twice or the server restarts mid-batch.	P0
FR-6.9	Quiet hours: no messages sent between 22:00 and 08:00 local; anything due in that window is deferred to 08:00.	P0
FR-6.10	Throttling: respect provider rate limits (e.g. 20 msg/sec WhatsApp, gateway-specific for SMS) with a token-bucket limiter and exponential-backoff retry (3 attempts: +1min, +10min, +1hr).	P0
FR-6.11	Delivery webhooks from each provider update message_job.status (sent → delivered → read/failed).	P0
FR-6.12	Thank-you message sent the day after each event to guests who accepted.	P2
FR-6.13	Admin can send a free-text broadcast to a filtered audience (e.g. “venue parking has changed”).	P1
FR-6.14	Opt-out: replying STOP on SMS/WhatsApp marks the guest do_not_contact; they still keep their working link.	P1

ID	Requirement	Priority
FR-6.15	Dry-run mode – render and log messages without actually sending – for pre-launch testing.	P0

5.7 QR codes and printed cards

ID	Requirement	Priority
FR-7.1	Generate a QR for each event's open link, at error-correction level H so it still scans if the print is decorative or partially obscured.	P0
FR-7.2	Provide the QR as SVG (vector, for the printer) and PNG at 1024px. Minimum recommended print size 2.5 cm × 2.5 cm stated in the UI.	P0
FR-7.3	Optional per-guest QR encoding <code>/i/{token}</code> plus a printable sheet (name + QR) for personalised cards.	P1
FR-7.4	QR destination URLs should be short (<code>domain.com/e/walima</code>) for a low-density, easy-to-scan code.	P0
FR-7.5	A human-readable fallback under the QR: "Or visit <code>domain.com/rsvp</code> and enter code A7K2M9 ".	P1
FR-7.6	Scans through the QR are tagged <code>invited_via = print_qr</code> so the host can see how well printed cards performed.	P1

6. User Journeys

6.1 Tokenized guest (WhatsApp)

1. Guest receives a WhatsApp message: "Assalamu Alaikum Rahim bhai – Murad & X's Walima, 12 Sept. Please confirm: link"
2. Taps the link → animated card opens, envelope unfolds, their name appears.
3. Scrolls: date, countdown, venue, map.
4. Taps **Accept Invitation** → form slides in, name and phone already filled.
5. Sets "guests attending = 2", taps Confirm.
6. Confetti + "See you on 12 September!" + Add to Calendar.
7. Receives a WhatsApp confirmation with the cancel link.
8. Gets reminders 15, 7 and 2 days before.

6.2 Printed-card guest (QR)

1. Receives a printed card with a QR.
2. Scans → opens `/e/walima` → same animated card, generic greeting.
3. Taps **Accept** → form asks name, phone, email, party size, and which events.
4. Submits → system creates the guest and invitation, tags source `print_qr`.
5. From here identical to 6.1.

6.3 Guest who cancels

1. Two days before the event, gets a reminder containing “Can’t make it? Cancel here”.
2. Taps → sees “You are confirmed for Walima, 12 Sept, 2 guests. Cancel this RSVP?”
3. Taps **Yes, cancel** → status cancelled, headcount drops by 2 immediately.
4. Sees “We’ll miss you — changed your mind? Re-accept”.
5. Future reminders for that invitation are cancelled.

6.4 Admin

1. Logs in, uploads a CSV of 800 guests, maps columns, fixes 12 flagged rows.
2. Tags guests by event (Mehedi 200, Marriage 800, Walima 600).
3. Goes to Send → Marriage → All → WhatsApp + SMS fallback → previews → sends.
4. Watches the dashboard fill up over the next 3 days.
5. Enables reminders for all three events.
6. Two days before, exports the “Caterer list” CSV: 612 accepted, 934 total headcount.

7. Technical Architecture

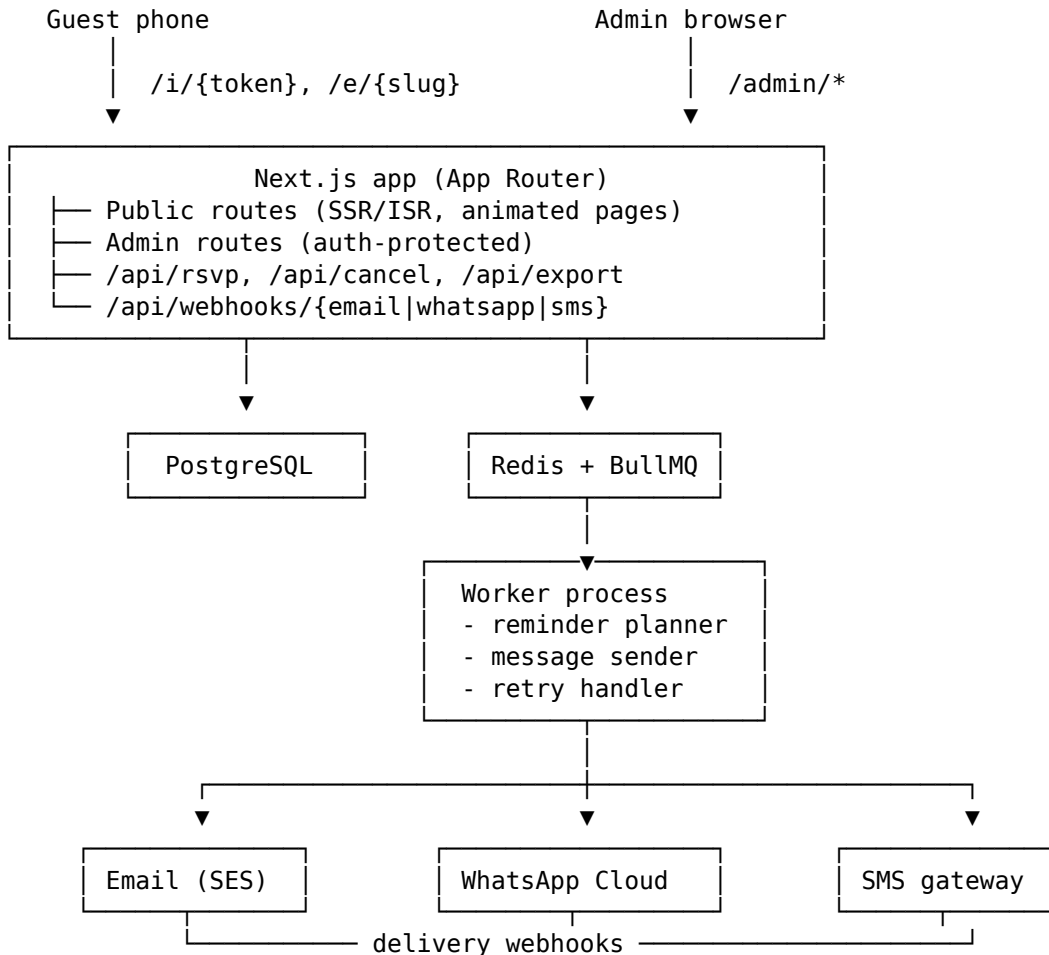
7.1 Stack

Layer	Choice	Why
Frontend + Backend	Next.js 15 (App Router) , TypeScript	One codebase for the public animated pages and the admin dashboard; server components make the invitation page fast
Styling Animation	Tailwind CSS + shadcn/ui Framer Motion for UI transitions, GSAP + ScrollTrigger for the card reveal, Lottie for illustrated flourishes, canvas-confetti for the success moment	Rapid, consistent admin UI Best balance of quality and mobile performance
Database	PostgreSQL 16	Relational data, strong constraints, easy CSV export
ORM Queue / Scheduler	Prisma BullMQ + Redis (or Postgres-backed pg-boss if Redis is undesirable)	Type-safe queries and migrations Delayed jobs, retries, rate limiting — exactly what reminders need
Auth (admin)	Auth.js (NextAuth) credentials provider + argon2id	Simple, self-hosted
File/Media	S3-compatible object storage (Cloudflare R2 / DigitalOcean Spaces) + CDN	Cheap image and audio hosting
Email WhatsApp	Resend or Amazon SES Meta WhatsApp Business Cloud API (direct) or Twilio/360dialog (reseller)	Good deliverability, low cost Template messages required for the first contact
SMS	Local BD gateway (e.g. bulksmsbd / Alpha Net / SSL Wireless) with a masking sender ID	Local delivery reliability and price
Hosting	Vercel (app) + Neon/Supabase (Postgres) + Upstash (Redis) — or a single VPS with Docker Compose	Vercel is simplest; VPS is cheaper and keeps data local

Layer	Choice	Why
Monitoring	Sentry (errors) + Better Stack / UptimeRobot (uptime) + structured logs	

Hosting note: if serverless (Vercel) is used, the queue worker must run somewhere always-on (Railway, Fly.io, or a small VPS) because serverless functions cannot hold long-running workers. A single VPS running everything in Docker Compose is the simplest and cheapest option for a one-off wedding.

7.2 System diagram



7.3 The reminder engine (most important piece)

Two cooperating jobs:

A. Planner — runs hourly (cron)

1. Find all published events with `starts_at` in the future.
2. For each enabled reminder_schedule, compute `send_at = starts_at - offset_days`, set to `send_at_local_time` in Asia/Dhaka, converted to UTC.
3. If `send_at` is in the past → skip (never send a late reminder).
4. Select the audience (accepted / pending / all).
5. For each invitation × channel, `INSERT ... ON CONFLICT (idempotency_key) DO NOTHING` into `message_job` with `scheduled_for = send_at`.
6. Enqueue each new job into BullMQ with the corresponding delay.

B. Sender — consumes the queue

1. Re-check the invitation's live status. If cancelled or declined (and audience was accepted), mark the job skipped and stop. *This is what makes cancellations take effect.*
2. Apply quiet hours: if now is 22:00–08:00 local, re-delay to 08:00.
3. Render the template with the guest's locale and placeholders.
- 4.

Pass through the channel rate limiter, then call the provider. 5. Store `provider_message_id`, set status sent. 6. On failure: retry up to 3 times with backoff; after that, try the next channel in the fallback chain; if all fail, mark failed and surface it in the admin message log.

Why idempotency keys matter: without `idempotency_key = invitation_id:schedule_id:channel`, a server restart or a duplicated cron run would message 800 guests twice. This single unique constraint is the cheapest insurance in the whole system.

7.4 Key API endpoints

Public | Method | Path | Purpose | | GET | `/i/{token}` | Render personalised invitation (server-rendered) | | GET | `/e/{eventSlug}` | Render open invitation | | POST | `/api/rsvp` | Submit acceptance (token or open) | | POST | `/api/rsvp/decline` | Record decline | | GET | `/i/{token}/cancel` | Cancellation confirmation page | | POST | `/api/rsvp/cancel` | Perform cancellation | | GET | `/api/ics/{token}` | Download calendar file | | POST | `/api/webhooks/whatsapp` | Delivery/read receipts (signature-verified) | | POST | `/api/webhooks/email` | Bounces, opens | | POST | `/api/webhooks/sms` | DLR callbacks |

Admin (auth required) | Method | Path | Purpose | | GET | `/api/admin/stats?eventId=` | Dashboard counts | | GET/POST/PATCH/DELETE | `/api/admin/guests` | Guest CRUD | | POST | `/api/admin/guests/import` | CSV import | | GET | `/api/admin/export?format=csv&filters=` | Streamed export | | POST | `/api/admin/invitations/send` | Send/queue invitations | | GET | `/api/admin/messages` | Message log | | POST | `/api/admin/messages/retry` | Retry failed | | GET/PATCH | `/api/admin/events/{id}` | Event content & settings | | GET/PATCH | `/api/admin/reminders` | Reminder schedule config | | GET | `/api/admin/qr/{eventId}?format=svg` | QR download |

7.5 Performance requirements

Requirement	Target
Invitation page LCP on 4G mobile	< 2.0 s
Invitation page on 3G	< 2.5 s
Total page weight (initial)	< 800 KB incl. hero image
Animation frame rate	60 fps on a mid-range Android (transform/opacity only — never animate width, height, top, left)
Images	WebP/AVIF, responsive srcset, lazy-loaded below the fold
Audio	Loaded only on user interaction
RSVP submit round-trip	< 500 ms p95
Dashboard load with 1,500 guests	< 1.5 s (paginated 50/page, indexed queries)
Message throughput	≥ 1,000 messages / 10 minutes
Respect prefers-reduced-motion	Animations reduced to simple fades

7.6 Security and privacy

Area	Requirement
Token strength	128-bit crypto-random, URL-safe base64 (22 chars). Not guessable, not sequential.
Token exposure	Treat as a bearer secret; do not log full tokens; strip from Referer via Referrer-Policy: no-referrer.
Enumeration	<code>/i/{invalid}</code> returns a generic “invitation not found” page — never reveals whether a token exists.
Cancellation safety	POST-only with CSRF token and explicit confirmation, so link scanners cannot cancel an RSVP.
Rate limiting	Per-IP and per-token limits on RSVP, cancel, and open-link submissions.

Area	Requirement
Bot protection	Cloudflare Turnstile on the open/QR form only (keeps the tokenized flow frictionless).
Transport	HTTPS everywhere, HSTS, secure + httpOnly + sameSite cookies.
Admin auth	argon2id password hashing, optional TOTP 2FA, 12-hour sessions, lockout after 5 failed logins.
Secrets	Provider API keys in environment variables / a secret manager — never in the database in plaintext, never in the repo.
PII minimisation	Store only name, email, phone. No addresses, no ID numbers.
Data retention	Auto-purge guest PII 90 days after the last event (configurable); export first if the host wants a keepsake.
Export auditing	Every CSV export logged with admin identity, filter used, and row count.
Webhook verification	Verify provider signatures (Meta X-Hub-Signature-256, SES SNS signature) before processing.
Backups	Nightly Postgres dump to object storage, 30-day retention, restore tested once before launch.
Consent	Form shows a one-line notice: “Your details will be used only for this wedding and deleted afterwards.”

7.7 Reliability

- Queue jobs survive restarts (persisted in Redis with AOF, or Postgres if using pg-boss).
- Planner is idempotent — safe to run every hour forever.
- Health endpoint `/api/health` checks DB, Redis, and provider reachability.
- Alert the admin by email if > 5% of a send wave fails.
- Graceful degradation: if WhatsApp is down, the fallback chain still delivers by SMS.

8. Integrations

8.1 WhatsApp (highest impact, most setup effort)

- Requires a **Meta Business account**, a verified business, a phone number dedicated to the WhatsApp Business API, and **pre-approved message templates** (approval typically 1–24 hours, sometimes longer).
- The first message to a guest **must** be a template message. Templates with a URL button work well: body text + “View Invitation” button.
- Once the guest replies, a 24-hour customer-service window opens in which free-form messages are allowed.
- Costs are per-conversation (utility/marketing category), roughly BDT 0.6–1.5 in Bangladesh at the time of writing — **verify current pricing before budgeting**.
- **Plan template approval at least 2 weeks before the first send.** This is the single most common schedule risk in this project.
- Needed templates: `invitation_v1`, `reminder_15_v1`, `reminder_7_v1`, `reminder_2_v1`, `cancel_confirm_v1` — each in Bangla and English.

8.2 SMS

- Use a Bangladeshi gateway with a masking/branded sender ID (needs BTRC-related paperwork through the provider; allow 3–7 days).

- **Watch the character count:** Bangla SMS uses UCS-2 encoding = only **70 characters per segment** (vs 160 for GSM-7 English). A Bangla reminder with a link can easily hit 3 segments and triple the cost. Recommendation: send SMS in transliterated or English text, keep the link short, and target ≤ 2 segments.
- SMS is the reliable fallback when WhatsApp fails or the number has no WhatsApp.

8.3 Email

- Transactional provider with a verified sending domain: **SPF, DKIM, and DMARC records must be configured** or invitations land in spam.
- Responsive HTML email template with the couple's branding, a hero image, a prominent button, and a plain-text alternative.
- Handle bounces and complaints via webhook; mark hard-bounced addresses invalid so reminders stop wasting sends.

8.4 Integration abstraction

Define one interface so channels are swappable:

```
interface MessageProvider {
  channel: 'email' | 'whatsapp' | 'sms';
  send(input: {
    to: string;
    templateKey: string;
    locale: 'bn' | 'en';
    variables: Record<string, string>;
    idempotencyKey: string;
  }): Promise<{ providerMessageId: string; status: 'sent' | 'queued' }>;
  parseWebhook(payload: unknown): DeliveryUpdate[];
}
```

This means swapping SMS vendors later is a one-file change, not a rewrite.

9. Design and Content Requirements

9.1 Visual direction per event

Event	Mood	Palette	Motion idea
Mehedi	Playful, festive, floral	Deep green, marigold yellow, terracotta	Mehedi/henna pattern draws itself along an SVG path as you scroll
Marriage Ceremony	Elegant, formal, luminous	Ivory, deep red, gold foil	Envelope opens, card slides out, gold-foil shimmer on the names
Walima	Warm, celebratory, modern	Blush pink, champagne, soft white	Curtain/petal reveal with a soft parallax gallery

9.2 Animation guidelines

- Intro animation ≤ 3 seconds, with a skip control.
- Content must be readable even if animations fail — never gate text behind JavaScript.
- Animate only transform and opacity for GPU compositing.
- Respect prefers-reduced-motion: reduce.
- Preload the hero asset; lazy-load gallery images.
- Test on a low-end Android device before launch, not just a desktop browser.

9.3 Content checklist per event

Couple names - event title (Bangla + English) - date and day - start time (and end if relevant) - venue name and full address - Google Maps link - host contact number - dress code (optional) - a short welcome message - 3-8 photos - optional background music track (licensed).

9.4 Accessibility

- WCAG 2.1 AA contrast on all text over images.
 - All interactive elements keyboard-reachable with visible focus rings.
 - Alt text on images; aria-live announcement on RSVP success.
 - Minimum 44×44px tap targets.
 - Base font size ≥ 16px (older relatives will be reading this).
-

10. Delivery Plan

Phase 0 – Setup (Week 1)

Repo, Next.js + Prisma + Postgres scaffold, CI, staging environment, domain purchase, and **start WhatsApp business verification immediately** (long lead time).

Phase 1 – Core RSVP (Weeks 2-3) – P0

Data model and migrations - token generation - one invitation page template with animation - accept/decline flow and gated form - open/QR flow with duplicate merging - cancellation flow - basic admin login and guest table.

Phase 2 – Admin & Export (Week 4) – P0

Full dashboard with per-event counts - guest CRUD - CSV import with validation - CSV export with filters - QR generation.

Phase 3 – Messaging (Weeks 5-6) – P0

Provider abstraction - email integration - SMS integration - WhatsApp integration - template management - send screen with preview and cost estimate - message log - delivery webhooks.

Phase 4 – Automation (Week 7) – P0

Reminder planner and sender - quiet hours - idempotency - retries and fallback chain - cancellation-aware skipping - reminder settings UI.

Phase 5 – Polish (Week 8) – P1

Three themed animated templates - Bangla/English toggle - content editor with live preview - trend charts - accessibility and performance pass.

Phase 6 – Hardening & Launch (Week 9)

Load test with 1,500 dummy guests - dry-run full reminder cycle with dates shifted - security review - backup and restore test - soft launch to 20 friendly guests - then full send.

Minimum viable path if time is short: Phases 1, 2 and 3 (email + SMS only) give a working system. WhatsApp and automated reminders can follow, with reminders sent manually as broadcasts in the meantime.

11. Testing Requirements

Type	Coverage
Unit	Phone normalisation, token generation, reminder date math (incl. timezone and DST edge cases), template rendering, CSV escaping
Integration	RSVP submit → DB → confirmation queued; cancel → status change → future jobs skipped; CSV import with duplicates and malformed rows
E2E (Playwright)	Full guest journey on mobile viewport for each of the three events; open-link/QR journey; cancellation journey; admin login → import → send → export
Provider	Sandbox/dry-run mode for all three channels; webhook signature verification; simulated failure and fallback
Load	1,500 guests, 3,000 queued messages, 200 concurrent RSVP submissions
Device	Real low-end Android, iPhone SE, and one desktop browser; test QR scan with the default camera app of both platforms
Print	Scan the actual printed QR at final size under normal indoor light before mass printing
Manual pre-launch checklist	Send every template to the host's own phone and inbox in both languages

12. Risks and Mitigations

#	Risk	Impact	Likelihood	Mitigation
1	WhatsApp template approval delayed or rejected	High	Medium	Start verification in week 1; submit conservative, utility-style templates; keep SMS+email as full fallback
2	Reminder fires twice or not at all	High	Medium	Idempotency key + hourly idempotent planner + dry-run rehearsal with shifted dates
3	Bangla SMS triples in cost due to UCS-2 encoding	Medium	High	Show a live segment/cost estimate in the send screen; prefer English/transliterated SMS; keep links short

#	Risk	Impact	Likelihood	Mitigation
4	Emails land in spam	Medium	High	SPF + DKIM + DMARC, warm the domain, avoid spam-trigger wording, include a plain-text part
5	Token link forwarded to others, causing duplicate or wrong RSVPs	Medium	Medium	Show the invitee's name on the page; open-link path exists for forwarders; admin can spot anomalies in the log
6	Elderly guests cannot use the digital flow	Medium	High	Keep the phone number prominent; admin can mark RSVPs manually on their behalf
7	Heavy animation makes the page slow on cheap phones	Medium	Medium	Performance budget, transform/opacity only, test on real low-end hardware, reduced-motion support
8	Traffic spike right after a bulk send crashes the site	High	Medium	Server-render + cache the static parts, CDN, autoscale or a right-sized VPS, load test first
9	Guest PII leaked via a guessable link or an unprotected export	High	Low	128-bit tokens, generic not-found page, auth-gated exports, audit logging
10	Event date changes after invitations are sent	Medium	Medium	Editing starts_at re-plans all future reminder jobs and offers a "date changed" broadcast
11	Accidental cancellation by a link-preview crawler	Medium	Medium	POST-only cancellation with explicit confirmation
12	SMS masking/sender-ID paperwork delays	Medium	Medium	Apply in week 1; non-masked numeric sender as an interim fallback

13. Open Questions

1. What are the exact dates and venues for the three events? (Needed to validate the reminder windows.)
 2. Expected guest count per event, and how much overlap between them?
 3. Are Mehedi and Walima invitations restricted subsets, or is everyone invited to everything?
 4. Should plus-ones be named individually, or is a count sufficient? (v1 assumes a count.)
 5. Is the primary language Bangla, English, or both — and which is the default?
 6. Do you want gift-registry, seating, or photo-sharing features later? (Affects data model choices now.)
 7. What is the budget ceiling for messaging? (Drives whether WhatsApp goes to all guests or only those without SMS.)
 8. Will you host on Vercel (easiest) or a VPS (cheapest, keeps data local)?
 9. Is a custom domain already owned, and what should it be?
 10. Should the site stay live as a memento after the wedding, or be taken down when PII is purged?
-

14. Appendix

14.1 Glossary

Term	Meaning
RSVP	The guest's reply — French <i>répondez s'il vous plaît</i> , "please reply"
Token	The long random string in a personal link that identifies one guest for one event
Idempotency	Doing the same operation twice produces the same result as doing it once — here, no duplicate messages
Webhook	The messaging provider calling <i>our</i> server to report delivery status
E.164	The international phone format, e.g. +8801712345678
UCS-2	The encoding SMS uses for Bangla; only 70 characters per message segment
LCP	Largest Contentful Paint — how long until the main content appears
Quiet hours	A time window in which the system refuses to send messages

14.2 CSV import template

full_name,email,phone,whatsapp,side,group_tags,events,max_guests,locale,preferred_channel
Rahim Uddin,rahim@example.com,01712345678,01712345678,groom,"family,dhaka","marriage,walima",4,bn,wha
Ayesha Karim,ayasha@example.com,01898765432,,bride,"university","mehedi,marriage",2,en,auto

Rules: phone accepts local or E.164 and is normalised; events is a comma-separated subset of mehedi,marriage,walima; blank whatsapp defaults to phone; rows with neither phone nor email are rejected with a reason.

14.3 Sample message copy

WhatsApp invitation (EN) > Assalamu Alaikum {{guest_name}}, > You are warmly invited to the {{event_title}} of {{couple_names}} on {{event_date}} at {{venue}}. > Please confirm your presence: {{invite_url}}

Reminder T-2 (EN) > {{guest_name}}, just 2 days to go! {{event_title}} — {{event_date}}, {{event_time}} at {{venue}}. > Directions: {{map_url}} > Can't make it? {{cancel_url}}

SMS reminder T-7 (short, 1 segment) > {{couple_names}} {{event_title}} in 7 days: {{event_date}}, {{venue}}. Details/cancel: {{invite_url}}

14.4 Environment variables

DATABASE_URL=

REDIS_URL=

NEXTAUTH_SECRET=

APP_BASE_URL=

TOKEN_PEPPER=

EMAIL_PROVIDER_API_KEY=

EMAIL_FROM=

WHATSAPP_PHONE_NUMBER_ID=

WHATSAPP_ACCESS_TOKEN=

WHATSAPP_WEBHOOK_VERIFY_TOKEN=

WHATSAPP_APP_SECRET=

SMS_API_KEY=

SMS_SENDER_ID=

SMS_BASE_URL=

S3_ENDPOINT= / S3_BUCKET= / S3_ACCESS_KEY= / S3_SECRET_KEY=

TURNSTILE_SITE_KEY= / TURNSTILE_SECRET_KEY=

SENTRY_DSN=

QUIET_HOURS_START=22:00

QUIET_HOURS_END=08:00

DRY_RUN=false

14.5 Requirements traceability

Original requirement	Covered by
Send animated invitation card to guests	FR-1.3, FR-1.4, §9.1, §9.2
Guests get a link redirecting to a hosted website	FR-1.1, FR-1.2, §7.1
Dynamic landing page with animations	FR-1.3–FR-1.14
Button to accept the invitation	FR-2.1, FR-2.6
Multiple invitations: Mehedi, Marriage, Walima	§4.2 event . type, FR-1.10, §9.1
Form collects name, email, phone	FR-2.2–FR-2.4
Form appears only after acceptance	FR-2.1
Admin dashboard with headcounts per event	FR-4.2, FR-4.3
Accepted vs pending counts	FR-4.2, §4.3
Download invitee list as CSV	FR-5.1–FR-5.7
Follow-ups at 15, 7 and 2 days before	FR-6.5, §7.3
Email, WhatsApp and SMS gateways	FR-6.1, FR-6.2, §8
Cancellation link in follow-up messages	FR-3.1–FR-3.7
QR code on printed cards linking to the app	FR-7.1–FR-7.6, FR-1.2