Ensure you have the following installed on your local machine: Ruby (use the same version as your project) Rails Git Heroku CLI (Install guide) PostgreSQL (if using a local database) 2. Prepare Your Application A) Set the Database to PostgreSQL Heroku supports PostgreSQL by default. Update your Gemfile: ruby-> gem 'pg' # Ensure this is added and `sqlite3` is removed (if present) Run-> bundle install Update your config/database.yml to use PostgreSQL in all environments. Example: default: &default adapter: postgresql encoding: unicode

pool: <%= ENV.fetch("RAILS_MAX_THREADS") { 5 } %>

1. Prerequisites

```
username: <%= ENV['DB_USERNAME'] %>
password: <%= ENV['DB_PASSWORD'] %>
 host: <%= ENV['DB_HOST'] || 'localhost' %>
development:
<<: *default
database: myapp_development
test:
<<: *default
database: myapp_test
production:
<<: *default
database: myapp_production
username: myapp
password: <%= ENV['MYAPP_DATABASE_PASSWORD'] %>
B)Add a Production Secret Key
Ensure you have a SECRET_KEY_BASE for production by generating it:
rake secret
```

Add it to Heroku later as an environment variable.

C) Update the Gemfile for Production
Ensure you include rails_12factor for serving static files:
gem 'rails_12factor', group: :production
3. Initialize Git Repository
If your project isn't already in a Git repository:
git init
git add .
git commit -m "Initial commit"
4. Login to Heroku
Log in to Heroku CLI:
heroku login
5. Create a Heroku App
Run: heroku create
This creates a new Heroku app with a unique URL (e.g., https://your-app-name.herokuapp.com).
6. Push Your Code to Heroku
Deploy your code to Heroku: git push heroku main
7. Setup the Database
Run migrations on the Heroku server: heroku run rake db:migrate

If you need seed data: heroku run rake db:seed
8. Add Environment Variables
Set your SECRET_KEY_BASE and other environment variables: heroku config:set SECRET_KEY_BASE= <your_generated_secret></your_generated_secret>
9. Test Your Application
Visit your application URL (e.g., https://your-app-name.herokuapp.com) in a browser.
10. Optional Add-Ons
Heroku Postgres: This is automatically added during setup. If needed, configure it:
heroku addons:create heroku-postgresql:hobby-dev
Logging: Use Heroku's logs for debugging:
heroku logstail
Next Steps
If you need custom domains or SSL, configure them via the Heroku dashboard.
Regularly push updates to Heroku using:
git push heroku main