

Product Requirements Document

Replit Vibe Coding PRD Generator

1. Product Overview

Product Name: Replit PRD Generator

Target Platform: Replit Agent

Primary User: First-time vibe coding students (beginners only)

Language: English only

Core Purpose: Transform student app ideas into Replit-optimized PRDs via Claude API, enabling successful first builds.

2. Problem Statement

Vibe coding students have ideas but submit vague prompts to Replit, resulting in:

1. Incomplete apps with missing features
2. Poor architecture decisions
3. Scope creep mid-build
4. Frustration and abandoned projects

This tool creates properly structured PRDs that Replit can execute against effectively.

3. Core User Flow

1. Student arrives at landing page
2. Student enters their app idea (free-form text)
3. App sends idea to Claude API with structured prompt
4. Claude returns detailed, Replit-optimized PRD
5. Student reviews PRD
6. Student copies PRD and pastes into Replit
7. Student builds their app

Total time: 2-5 minutes

4. Feature Requirements

4.1 Landing Page

Components:

1. Hero Section

- Headline: 'Turn Your App Idea Into a Clear Plan'
- Subheadline: 'Get a professional PRD in 60 seconds—ready for Replit'
- Primary CTA: 'Generate My PRD' (large button)

2. How It Works (3 steps)

- Step 1: 'Describe your app idea'
- Step 2: 'AI creates your PRD'
- Step 3: 'Build it in Replit'

3. Example Section

- 'See an Example' expandable section
- Shows: Student input → Generated PRD (truncated)
- Real example from successful student project

4. Trust Indicators

- 'Powered by Claude AI'
- 'Optimized for Replit'
- 'Used by [X] students' (if applicable)

4.2 Input Form

Single Page Form:

1. Main Input Field

- Large textarea (min 100 chars, max 2000 chars)
- Placeholder: 'Describe your app idea. What does it do? Who is it for? What problem does it solve?'
- Character counter: 'X / 2000 characters'
- Helpful prompt above field: 'Be specific! Include what your app does, who will use it, and why they need it.'

2. Example Prompt Toggle

- Checkbox: 'Show me an example'
- When checked, displays good vs bad examples:
 - Bad: 'A todo app'
 - Good: 'A todo app for college students that lets them add tasks by voice, set priorities, and get reminded before deadlines. It should work on mobile and sync across devices.'

3. Optional Enhancement Questions (Expandable)

- 'Want to add more details? (Optional)'
- When expanded, shows 3-5 quick checkboxes:
 - User accounts needed
 - Mobile-first design
 - Real-time features
 - File uploads
 - Dark mode
- These get appended to the prompt sent to Claude

4. Generate Button

- Primary button: 'Generate My PRD'
- Disabled until min character count met
- Loading state: 'Generating...' with spinner

4.3 Claude API Integration

Prompt Template Structure:

```
You are a product requirements document expert specializing in Replit projects for beginner developers.

A student has described their app idea below. Create a comprehensive PRD that is optimized for Replit Agent to build successfully.

STUDENT'S IDEA:
{user_input}

{optional_requirements}

Generate a PRD with these sections:

1. PROJECT OVERVIEW
2. PROBLEM & SOLUTION
3. CORE FEATURES (Must Have)
4. USER INTERFACE REQUIREMENTS
5. TECHNICAL SPECIFICATIONS FOR REPLIT
6. DATA MODEL
7. API REQUIREMENTS
8. USER STORIES
9. SUCCESS CRITERIA
10. OUT OF SCOPE (v1)
11. REPLIT IMPLEMENTATION NOTES

IMPORTANT GUIDELINES:
- Write for a beginner developer
- Be specific and detailed
- Use clear, simple language
- Make this immediately actionable in Replit
- Keep scope reasonable for a first project (can be built in 2-8 hours)
- Prioritize functionality over perfection
- If the idea is too complex, simplify it to a viable v1
```

API Configuration:

- Model: claude-sonnet-4-20250514
- Max tokens: 4000
- Temperature: 0.7

4.4 PRD Display Page

Layout:

1. Header

- 'Your PRD is Ready!'
- Subtext: 'Review your plan, then copy it to Replit'

2. PRD Content Area

- Full-width text display
- Properly formatted markdown rendering
- Sections clearly separated
- Readable typography (16px base, good line height)
- Light gray background, white content card

3. Action Buttons (*Sticky footer on mobile*)

- Primary: 'Copy PRD' (copies full markdown to clipboard)
- Secondary: 'Download as .md'
- Tertiary: 'Start Over'
- Success message on copy: '✓ Copied! Now paste this into Replit'

4. Quick Tips Section (*Collapsible*)

- 'How to use this PRD in Replit:'
- Step 1: Click 'Copy PRD' above
- Step 2: Go to Replit and create new project
- Step 3: Paste this PRD into the Replit Agent chat
- Step 4: Let Replit build your app!

5. Feedback Option

- Small link: 'PRD missing something? Start over'
- No ratings/surveys (keep it simple)

4.5 Error Handling

Error States:

1. Input Too Short

- Inline message: 'Tell us more! Add at least 100 characters describing your idea.'
- Generate button stays disabled

2. Input Too Long

- Inline message: 'Keep it under 2000 characters. Focus on the core idea.'
- Character count turns red

3. API Failure

- Modal: 'Oops! Something went wrong generating your PRD.'
- CTA: 'Try Again'
- Preserve user's input
- Log error for debugging

4. API Timeout

- After 30 seconds: 'This is taking longer than expected...'
- After 60 seconds: Show error, allow retry
- Don't lose user input

5. Rate Limiting

- Message: 'You've generated 5 PRDs today. Try again tomorrow or contact us for more.'
- Store count in localStorage (simple, no auth needed)

5. Technical Specifications

5.1 Frontend

Stack:

- Next.js 14+ (App Router)
- React 18+
- Tailwind CSS
- TypeScript

Key Libraries:

- react-markdown (for PRD display)
- react-copy-to-clipboard
- zod (input validation)

5.2 Backend

API Routes:

- POST /api/generate-prd
 - Accepts: user_input, optional_requirements
 - Returns: generated PRD (markdown string)
 - Rate limit: 5 requests per IP per day
 - Timeout: 60 seconds

Environment Variables:

```
ANTHROPIC_API_KEY=xxx  
RATE_LIMIT_MAX=5  
RATE_LIMIT_WINDOW=86400
```

5.3 Data Flow

1. User submits form
2. Client validates input (length, format)
3. POST to /api/generate-prd
4. Server checks rate limit
5. Server constructs prompt with template
6. Server calls Claude API

7. Server returns markdown PRD
8. Client renders PRD
9. User copies to clipboard
10. Client stores generation count in localStorage

No database required for v1

6. UI/UX Requirements

6.1 Design Principles

- Minimal and clean
- Large, readable text
- Obvious CTAs
- No overwhelming choices
- Mobile-first responsive
- Fast loading (<2s)

6.2 Color Palette

- Primary: Replit orange/red (#F26207 or similar)
- Secondary: Dark gray (#1C2333)
- Background: Light gray (#F5F5F5)
- Text: Dark (#1A1A1A)
- Success: Green (#10B981)
- Error: Red (#EF4444)

6.3 Typography

- Headings: Inter or similar sans-serif (bold)
- Body: Inter regular
- Code/PRD display: Fira Mono or monospace
- Minimum 16px base size
- 1.6 line height for readability

6.4 Responsive Breakpoints

- Mobile: 375px - 767px (single column)
- Tablet: 768px - 1023px (single column, larger text)
- Desktop: 1024px+ (max-width 800px for form, 900px for PRD)

7. Content & Copywriting

7.1 Tone

- Encouraging and supportive
- Clear and direct
- Assumes zero coding knowledge
- Avoids jargon
- Focuses on action ('Generate', 'Build', 'Create')

7.2 Key Copy Elements

Landing Page:

- H1: 'Turn Your App Idea Into a Clear Plan'
- Subhead: 'Get a professional PRD in 60 seconds—ready for Replit to build'
- CTA: 'Generate My PRD'

Input Form:

- Label: 'What's your app idea?'
- Placeholder: 'Describe your app idea. What does it do? Who is it for? What problem does it solve?'
- Example toggle: 'Show me an example'

Results Page:

- H1: 'Your PRD is Ready!'
- Subhead: 'Copy this to Replit and start building'
- Primary CTA: 'Copy PRD'
- Success message: 'Copied! Now paste this into Replit'

8. Success Metrics

8.1 Primary Metrics

1. PRD generation completion rate (target: >85%)
2. Average time to generate (target: <60 seconds)
3. Copy-to-clipboard usage (target: >90%)

8.2 Quality Metrics

1. Average PRD length (target: 1200-2000 words)
2. Input quality (average character count: target >300)
3. Error rate (target: <5%)

8.3 Engagement Metrics

1. Return users (target: >30% generate 2+ PRDs)
2. Download rate (target: >20%)
3. Bounce rate on input form (target: <30%)

9. Performance Requirements

1. Landing page load: <1.5s
2. Form interaction: Instant (<100ms)
3. API response: <30s (average <15s)
4. PRD render: <500ms
5. Copy to clipboard: Instant
6. Mobile performance score: >90 (Lighthouse)

10. Security & Privacy

1. Rate limiting: 5 PRDs per IP per 24 hours (localStorage + server-side)
2. Input sanitization: Strip HTML, limit characters
3. API key security: Server-side only, never exposed
4. No data storage: PRDs not saved (user downloads if needed)
5. HTTPS only
6. No analytics tracking (GDPR friendly)
7. No cookies (except essential)

11. Launch Criteria

Must Have:

- ✓ Landing page with example
- ✓ Input form with validation
- ✓ Claude API integration working
- ✓ PRD generation functional
- ✓ Copy to clipboard working
- ✓ Download as .md working
- ✓ Mobile responsive
- ✓ Error handling for all states
- ✓ Rate limiting functional
- ✓ <3s page loads

Should Have:

1. Good vs bad example toggle
2. Optional requirements checkboxes
3. Replit usage instructions
4. Proper loading states

Nice to Have:

1. Syntax highlighting in PRD display
2. Print-friendly PRD format
3. Share link (expires after view)

12. Out of Scope (v1)

- User accounts
- Saved PRDs
- PRD editing interface
- Multiple language support
- PRD versioning
- Collaborative features
- Templates
- Direct Replit API integration
- Payment/premium features
- Analytics dashboard
- Email delivery
- PRD comparison
- Community gallery

13. Future Enhancements (Post-v1)

1. 'Refine PRD' feature (send back to Claude with modifications)
2. Track success: 'Did Replit build this successfully?' feedback
3. Example gallery from real student projects
4. Direct Replit project creation (if API available)
5. Optional user accounts to save PRD history
6. PRD quality scoring
7. Common patterns detection (todo app, blog, etc.)

14. Development Phases

Phase 1: MVP (Week 1)

- Landing page
- Basic input form
- Claude API integration
- Simple PRD display
- Copy to clipboard

Phase 2: Enhancement (Week 2)

- Optional requirements checkboxes
- Example toggle
- Download functionality
- Error handling
- Rate limiting

Phase 3: Polish (Week 3)

- Mobile optimization
- Loading states
- Better PRD formatting
- Instructions section
- Performance optimization

Implementation Notes

This PRD is designed to be fed directly into Replit for implementation. The focus is on simplicity and ensuring first-time coders get PRDs that actually result in successful builds.

All decisions have been finalized:

- No user accounts or saved drafts
- Replit-optimized output only
- Beginner mode only
- English language only

- Claude API generates PRDs from raw student input