# Specificații Tehnice și de Implementare V18.1: Showcase, Teme & Onboarding

**Document de Referință pentru Dezvoltare** **Ultima Versiune Integrată:** 18.1 (28 iunie 2025)

## Partea I: Implementarea "Showcase-ului de Unelte Gratuite" (Prioritate pentru V1.0)

**Obiectiv:** Crearea unei secțiuni publice pe site-ul principal care oferă unelte "lite" pentru a atrage trafic, a demonstra valoarea și a converti vizitatorii în utilizatori premium.

### 1.1. Arhitectura "Modului Dual" pentru Generatoare

Fiecare Component Generator (fost AiPlugin) trebuie să poată opera în două moduri:

* **Mod Complet:** În interiorul studioului, pentru utilizatorii autentificați, cu acces la toate opțiunile.
* **Mod Lite:** Public, fără autentificare, cu funcționalități limitate.

**Instrucțiuni de Implementare (Backend):** Refactorizați funcția de bază a fiecărui generator (ex: generateWorksheetComponent) pentru a accepta un parametru mode: 'full' | 'lite'. Dacă modul este 'lite', funcția va aplica limitări (ex: o singură pagină, adăugare de watermark).

### 1.2. Backend: Endpoint-uri Publice

* **Acțiune:** Pentru fiecare unealtă din showcase, se va crea un nou endpoint public pe API Gateway.
* **Securitate:** Aceste endpoint-uri vor avea un **rate-limiter** strict (ex: max 5 cereri pe oră de la o singură adresă IP) pentru a preveni abuzul.
* **Exemplu:** Un nou endpoint POST /v1/public/generate-coloring-page care va apela WorksheetModule în mode: 'lite'.

### 1.3. Frontend: Construcția Paginii "Showcase"

* **Acțiune:** Creați o nouă pagină pe site-ul public (/unelte-gratuite).
* **Design:** O grilă atractivă de "carduri". Fiecare card reprezintă o unealtă gratuită (ex: "Generator de Pagini de Colorat").
* **Flux:**
  1. Utilizatorul dă click pe un card.
  2. Se deschide o interfață simplă, dedicată acelei unelte (nu întregul studio).
  3. Utilizatorul introduce un prompt (ex: "un astronaut pe lună").
  4. Apasă "Generează".
  5. Apare rezultatul (imaginea paginii de colorat) și un buton de descărcare.
  6. Alături, apare un mesaj de marketing clar: *"V-a plăcut? Transformați această pagină într-o poveste completă în studioul nostru premium."*

### 1.4. Exemplu de Prompt de Implementare (Pentru AI Assistant/Dezvoltator)

"Create the public-facing interface for the 'Lite Coloring Page Generator'. It should be a single React component with:  
1. A text input for the user's theme (e.g., 'a friendly dinosaur').  
2. A 'Generate' button that shows a loading state.  
3. An area to display the resulting image.  
4. A 'Download' button that becomes active when the image is ready.  
5. This component will call the public API endpoint `POST /v1/public/generate-coloring-page` with the theme as the payload."

## Partea II: Implementarea Conținutului Pre-generat și Onboarding (Prioritate pentru V1.0)

**Obiectiv:** Îmbunătățirea retenției noilor utilizatori prin eliminarea "paginii goale" și oferirea de valoare instantanee.

### 2.1. Crearea Atelierelor "Master"

* **Acțiune:** Folosind propriul nostru AI Play & Learn Studio, noi (ca administratori) vom crea 3-5 ateliere "perfecte". Acestea vor fi salvate și marcate în baza de date cu un flag special, de exemplu is\_template: true.

### 2.2. Mecanismul de "Clonare" la Înregistrare

* **Acțiune:** Modificați logica de înregistrare a unui utilizator nou.
* **Flux (Backend):**
  1. După ce un utilizator nou este creat cu succes în tabelul users, se declanșează un nou proces.
  2. Sistemul caută în tabelul workshops toate înregistrările cu is\_template: true.
  3. Pentru fiecare template găsit, se creează o **copie completă** (un "deep clone") și se atribuie noului user\_id.
  4. **Rezultat:** Când utilizatorul se loghează pentru prima dată, el găsește deja aceste 3-5 ateliere în tabloul său de bord, ca și cum le-ar fi creat el însuși.

### 2.3. Exemplu de Prompt de Implementare (Pentru Backend)

"After a new user is successfully created, trigger a 'seedUserAccount' function. This function should:  
1. Query the `workshops` table to find all records where `is\_template = TRUE`.  
2. Iterate through each template workshop.  
3. For each template, create a new record in the `workshops` table, duplicating all its data (title, theme, components\_json, etc.).  
4. Assign the `user\_id` of the new user to this new record.  
5. Ensure this process is asynchronous and does not block the user's registration."

## Partea III: Implementarea Sistemului de Teme Vizuale (Prioritate pentru V1.0 - Bază)

**Obiectiv:** Permiterea utilizatorilor să personalizeze aspectul vizual al produselor lor finale cu un singur click.

### 3.1. Definirea unei "Teme"

* **Acțiune:** O "Temă" va fi un obiect JSON stocat în baza de date. Acesta va conține un set de variabile de stil.
* **Exemplu theme\_jungle.json:**  
  {  
   "name": "Aventură în Junglă",  
   "colors": {  
   "primary": "#34D399", // verde smarald  
   "secondary": "#FBBF24", // galben chihlimbar  
   "text": "#374151"  
   },  
   "fonts": {  
   "headings": "'Luckiest Guy', cursive",  
   "body": "'Open Sans', sans-serif"  
   },  
   "backgroundImageUrl": "gs://bucket/themes/jungle\_pattern.svg"  
  }

### 3.2. Aplicarea Temei (Backend)

* **Acțiune:** Când se generează pagina web (index.html) și "Hub-ul PDF", sistemul va citi obiectul JSON al temei selectate.
* **Implementare:**
  + **Pentru Pagina Web:** Va injecta variabilele de stil într-un tag <style> în antetul paginii HTML, sub formă de variabile CSS. (ex: --color-primary: #34D399;)
  + **Pentru PDF:** Biblioteca de generare PDF va fi instruită să folosească culorile și fonturile specificate din obiectul temei.

### 3.3. Interfața de Selecție (Frontend)

* **Acțiune:** În **Editorul de Ateliere**, se adaugă o nouă secțiune în panoul de configurare.
* **Design:** O galerie de miniaturi. Fiecare miniatură arată o previzualizare a temei (ex: o paletă de culori și un exemplu de font) și are numele temei sub ea.
* **Funcționalitate:** La click pe o temă, **Previzualizarea în Timp Real** din dreapta își va actualiza instantaneu stilurile pentru a reflecta noua selecție.

### 3.4. Considerații pentru Add-on-uri (Viziune pe Termen Lung)

Arhitectura trebuie să fie modulară de la început. Component Generators trebuie tratați ca niște plugin-uri. Acest lucru va permite în viitor adăugarea de noi generatoare (plătite sau gratuite) fără a rescrie nucleul aplicației, ci doar prin adăugarea unui nou "plugin" la sistem.