

SaaS cost reduction checklist

At the intersection of innovation and fiscal responsibility lies the opportunity to reimagine how we approach software costs. This checklist serves as your compass for navigating the complex landscape of SaaS expenditure, empowering you to cultivate a tech ecosystem that drives growth while preserving financial health.

No	Activities	✓
01	<p>Strategic cost categorisation</p> <ul style="list-style-type: none"> ▶ Create distinct software expense categories that reflect your operational structure ▶ Assign all current tools to appropriate categories (e.g., collaboration, engineering, design) ▶ Calculate total spend per category to identify priority areas ▶ Establish quarterly review cycles for each category ▶ Designate category owners responsible for optimisation 	
02	<p>Tool consolidation & optimisation</p> <ul style="list-style-type: none"> ▶ Conduct a comprehensive audit of all active software subscriptions ▶ Identify redundant tools with overlapping functionality ▶ Survey team members about tool utilisation and satisfaction ▶ Eliminate tools with low adoption rates or minimal ROI ▶ Ensure remaining tools are being leveraged to their full potential 	
03	<p>Payment structure refinement</p> <ul style="list-style-type: none"> ▶ Review all subscription payment schedules ▶ Calculate potential savings of switching from monthly to annual plans ▶ Identify providers offering significant annual discounts ▶ Develop a transition schedule for converting strategic tools to annual plans ▶ Negotiate custom pricing for high-value, long-term commitments 	

04	<p>Client cost integration</p> <ul style="list-style-type: none"> ▶ Create a system for tracking per-client software utilisation ▶ Develop a formula for including software costs in client estimates ▶ Update proposal templates to reflect software as a project expense ▶ Establish transparency in how software costs affect project pricing ▶ Review client-specific tool requirements before project initiation 	
05	<p>Tool adoption protocol</p> <ul style="list-style-type: none"> ▶ Implement a standardised process for new tool requests ▶ Create an evaluation framework with clear criteria for adoption ▶ Define decision-makers for various software categories ▶ Establish mandatory trial periods for all new tools ▶ Require ROI projections before permanent adoption ▶ Document integration requirements with existing systems 	
06	<p>Licence management system</p> <ul style="list-style-type: none"> ▶ Create a central inventory of all software licences ▶ Track user-specific licences and their active status ▶ Integrate licence deactivation into the employee offboarding process ▶ Conduct monthly audits to identify and eliminate "ghost" accounts ▶ Implement just-in-time licence allocation for seasonal needs 	
07	<p>Continuous optimisation framework</p> <ul style="list-style-type: none"> ▶ Schedule quarterly comprehensive software expense reviews ▶ Monitor for unexpected cost increases across all platforms ▶ Track usage metrics to identify underutilised tools ▶ Reassess value alignment with business objectives ▶ Solicit team feedback on potential improvements 	
08	<p>Integration architecture design</p> <ul style="list-style-type: none"> ▶ Identify lead software platforms that serve as operational cornerstones ▶ Prioritise tools with robust API capabilities ▶ Map integration points between core systems ▶ Develop a visual representation of your software ecosystem ▶ Evaluate new tools primarily on their ability to integrate with lead software 	