PixConvert

Executive Feasibility Snapshot

Assessment of native two-app scope, offline image conversion, metadata privacy controls, and the deliberate exclusion of server-based video conversion

Product concept

PixConvert and PixConvert for Mac were created to solve a practical Apple-platform workflow gap: a fast, trustworthy way to convert images without sending personal files to dubious websites or adding unnecessary workflow friction.
Both apps convert images locally, support batch processing, reduce file size through format and resize settings, and offer optional metadata removal. The concept phase also tested whether video conversion or a more complicated combined macOS path truly fit the product.

Executive decision

Proceed with two separate native apps — one for iOS and one for macOS — with aligned purpose but platform-specific interaction models. Keep the product strictly offline and image-focused. Exclude server-based video conversion from the current scope.
Reason: building a separate macOS app was operationally cleaner than forcing desktop behavior into the existing iOS codebase, while a cloud video layer would weaken the privacy-first product promise, materially expand technical scope, and introduce recurring operating cost without proportional user value.

Separate native apps

iOS and macOS remain separate so each platform keeps its own natural import and export flow.
Assessment: lower integration complexity.

On-device image workflow

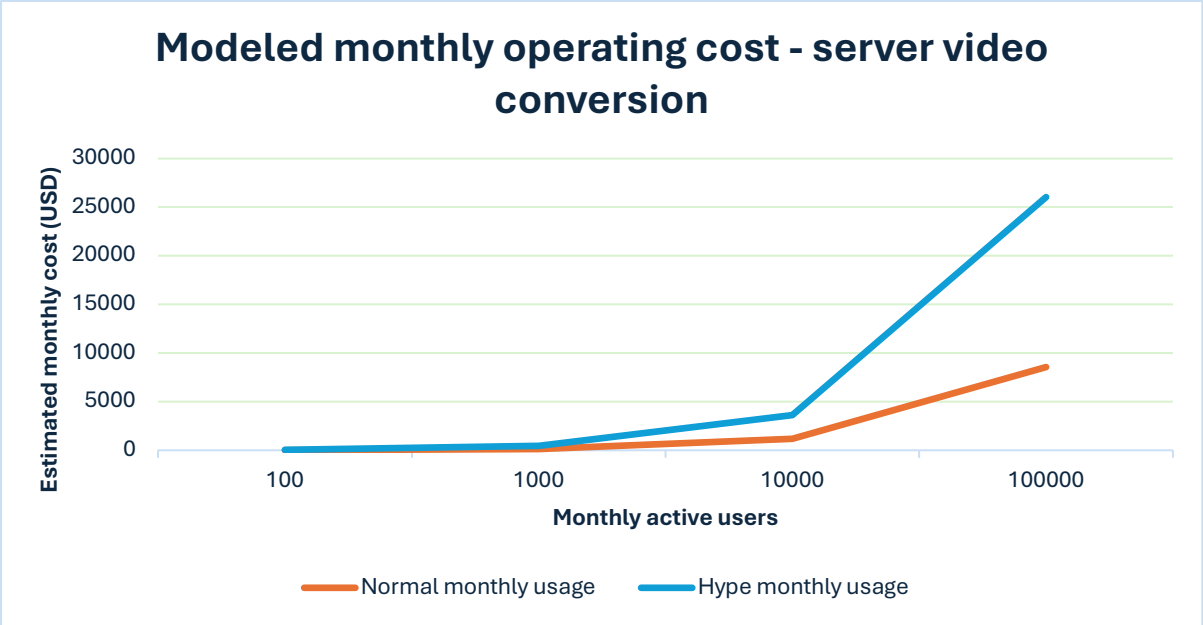
Images stay on device, and metadata can be stripped during export.
Assessment: matches the product's privacy purpose.

Rejected cloud video layer

About $26,040 modeled monthly transcoding-only cost at 100,000 monthly active users in the hype case.
Interpretation: recurring server burden without equal upside.

Cost-growth view

Server-based HD video conversion benchmark; the chart illustrates the recurring monthly cost that would have been introduced by adding cloud video processing to an otherwise offline utility.



Note: the plotted curve is a management illustration. It uses a simplified HD transcoding model and excludes storage, bandwidth, taxes, negotiated discounts, abuse tooling, and support overhead. Actual spend would vary with job length, traffic concentration, and chosen vendor architecture.

Benchmark model and management interpretation

Illustrative benchmark assumptions

- Illustrative vendor baseline: AWS Elemental MediaConvert Basic tier equivalent, modeled from current public HD output pricing.
- Benchmark reflects one 1080p HD output per conversion job.
- Normal case: 3 video exports per user / month at 3 minutes average = 9 output minutes.
- Hype case: 8 video exports per user / month at 4 minutes average = 32 output minutes.
- Transcoding-only view: storage, upload / download bandwidth, auth, abuse controls, support, and tax remain excluded.

Why the leverage becomes decisive

Image conversion on device mostly scales with local hardware and does not create a direct per-use server bill.
Server video conversion scales with processed output minutes, which turns growth in active users into recurring infrastructure spend.
The same feature would also require users to upload private media, wait for queuing and processing, and trust a cloud handling path that the product was intentionally designed to avoid.

Decision framework

Monthly server-cost benchmark, privacy / scope-risk interpretation, and management recommendation.

Monthly cost benchmark (USD)

Normal monthly usage benchmark **Hype monthly usage benchmark**

Monthly active users	Cloud transcoding cost	Monthly active users	Cloud transcoding cost
100	$13.50	**100**	$48
1,000	$135	**1,000**	$480
10,000	$1,174	**10,000**	$3,612
100,000	$8,560	**100,000**	$26,040

Annualized view at 100,000 monthly active users: about $102,720 per year in the normal benchmark and about $312,480 per year in the hype benchmark, before storage and transfer costs. Apple Developer Program membership remains $99 per year by comparison.

Risk and management interpretation

Risk area	Impact	Operational response	Expected outcome
User media privacy	High	Keep conversion local, avoid central media retention, and preserve optional metadata removal for export.	Lower data-exposure risk and a clearer trust story.
Product scope inflation	High	Keep PixConvert focused on image conversion rather than broadening into a heavier media suite.	Faster delivery and a more maintainable product.
Operating-cost volatility	Medium	Exclude cloud video processing from the current scope and keep infrastructure optional rather than foundational.	Near-zero recurring media-processing cost.
Cross-platform complexity	High	Use two native apps with aligned intent instead of forcing macOS behavior into the iOS code path.	Cleaner UX and fewer platform compromises.

Management recommendation

PixConvert should remain a privacy-first, offline image utility across iOS and macOS. The two-app structure is justified by lower engineering friction and better platform fit, while server-based video conversion should remain out of scope unless a future version gains clear monetization and a privacy model that users explicitly accept.
What this demonstrates: disciplined platform scoping, privacy-aware feature design, and cost-aware rejection of server dependency even when a feature is technically feasible.

Document basis: PixConvert iOS and macOS technical briefs plus project source files. Video-cost benchmark uses public AWS Elemental MediaConvert pricing; Apple distribution context was cross-checked against Apple Developer Program pricing.