SUNUCU TEKNİK

Yazılım Rehberi
Ürün Geliştirme

https://sunucuteknik.com bilgi@sunucuteknik.com
Çamçeşme Mah. Erva Sk. No:1 Pendik, İstanbul
+90 216 397 53 96

Bu rehberde, Sunucu Teknik, yazılım ürünü geliştirme projenize başlamadan önce düşünmeniz gereken her şeyi vurguluyor. Teorik detaylardan başlayıp pratik organizasyon ipuçlarına kadar her şeyi kapsıyoruz.

Yazılım Ürünü Geliştirme Sürecinin Önemli Yönleri

Kullanıcı Deneyimi (UX) Kritik Öneme Sahip

Kullanıcı Kişilikleri. Yazılımınızın kullanacak tüm insan profillerini listelemeniz ve bu insanların kullanım amaçlarını düşünmeniz gerekmektedir. Her bir kişiliğin ürünle nasıl etkileşimde olabileceğini hayal ederek, kullanıcı senaryoları oluşturursunuz. Bu çalışma, ürününüzü satın almaya yönlendirebilecek özellikleri görmek için size yardımcı olacaktır.

Kolay Kullanılabilir Arayüz. Herhangi bir senaryonun arkasındaki işlev ne kadar karmaşık olursa olsun, ürün kullanıcıya bu karmaşıklığı yansıtmamalıdır. Arayüzünüzün sade ve navigasyonunun anlaşılır olması önemlidir. Görsel çekicilik de önemlidir; başlangıçta kullanıcıların dikkatini çekerken, ürününüz hakkında uzun süreli olumlu izlenimler oluşturmaya da yatırım yapmalısınız.

Örneğin, bir e-ticaret platformu için yazılım geliştiriyorsanız, çeşitli kullanıcı profillerini (alıcılar, satıcılar, yöneticiler) ve bu kullanıcıların platformu nasıl kullanacaklarını düşünmelisiniz. Alıcılar ürünleri arayacak, inceleyecek ve satın alacaklar. Satıcılar ürünlerini listeleyecek, stokları yönetecek ve siparişleri işleyecekler. Yöneticiler ise platformun genel performansını izleyecek, kullanıcıları yönetecek ve gerekli ayarlamaları yapacaklar.
Kullanıcı yaşını da dikkate almalısınız. Örneğin, alıcılar arasında genç kullanıcılar daha teknolojiye hakim olabilirken, yaşlı kullanıcılar için daha basit ve anlaşılır bir arayüz sunmak önemli olabilir. Satıcılar için ise ürün listeleme ve stok yönetimi gibi işlevlerin kullanıcı dostu ve hızlı olması gerekebilir.
Görsel çekiciliğe de önem vermek gereklidir. Kullanıcıların ürünleri keşfederken dikkatlerini çekmek ve olumlu bir ilk izlenim bırakmak için kullanıcı arayüzünün estetik ve kullanıcı dostu olması önemlidir.

Talep Odaklı Özellik Seçimi. Eğer bir birinden farklı kullanıcı ihtiyaçlarını karşılamaya çalışırsanız, ürününüz çok karmaşık olabilir ve pazarda başarılı olamayabilir. Fonksiyonelliğinizi sınırlamak önemlidir, size iyi bir seçenek önerebiliriz :Minimum Viable Product (MVP) oluşturmaktır. Farklı kullanıcı kişilikleri için senaryolar arasında en yaygın olan özellikleri belirleyip, bunları ürününüzün başlangıç versiyonunun çekirdeği yapabilirsiniz.

Bir ürünü piyasaya sürdükten sonra oluşan hataları düzeltmek için indirilebilir güncellemeler veya bir sonraki sürüm sunmak, kullanıcıların genellikle ürününüz hakkında olumsuz bir izlenim edinmesine ve güvensizliklerine neden olabilir. Kaliteli ve güvenli bir ürün sunabilmek için ürün geliştirme sürecinizi Microsoft'un SDL gibi Güvenli Yazılım Geliştirme Yaşam Döngüsü ilkeleri etrafında şekillendirin. Bu yaklaşım, güvenlik ve kaliteyi odak noktanız yaparak, kullanıcılarınıza daha güvenilir bir deneyim sunmanızı sağlar. Bu sayede, ürününüzün piyasaya sürülmesi ve sonrasındaki süreçlerde oluşabilecek sorunları minimize etme şansınızı artırırsınız.

Sürekli Geliştirme ve Yayına Alma. Ürününüzün sürekli olarak gelişmesi, zamanla daha fazla kullanıcı senaryosunu kapsamanıza olanak tanır. Kullanıcıların ihtiyaçları ve gelecek ürün sürümleri için beklentileri hakkında bilgi almak için uygulama performansı izleme kullanmayı düşünün.

Teknolojik Yönler Dikkate Alınmalıdır

Her projeye uymayan ancak kişisel olarak düşünülebilecek bir dizi teknoloji stratejisi de bulunmaktadır.

Hizmet Olarak Yazılım (SaaS)

SaaS modeli, bulut sunucularda barındırılan yazılım ürününe online erişim sağlamayı içerir, bu da size ürün dağıtımıyla ilgili birçok sorunu ortadan kaldırır. Ayrıca, bu model müşteri kitlenizi belirli donanımlara veya platformlara daraltmaz; müşterinin yazılımınızı kullanabilmesi için ihtiyacı olan tek şey internet bağlantısı ve ürününüze abonelik yapmasıdır. Esnek fiyatlandırma seçenekleri belirleyebilme ve güncellemeleri anında teslim edebilme avantajınız da bulunmaktadır.

Uygulama İzleme (Application Monitoring)

Uygulama izleme, ürününüzün kullanımındaki performansını izlemenize ve müşteri davranışları hakkında veri elde etmenize olanak tanıyan bir stratejidir. Örneğin, kullanıcı performans sorunlarını anlamak ve çözmek için kod işlemlerini geri izleyebilirsiniz. Ayrıca, kullanıcıların yazılımınızla nasıl etkileşimde bulundukları hakkında raporlar alarak, ürününüzün kullanıcı ara yüzü ve işlevselliğinden ne beklediklerini daha iyi anlayabilirsiniz.

Donanımla Entegrasyon

Eğer şirketinizin donanımı için yazılım geliştirmeniz gerekiyorsa (IoT akıllı cihazlar, endüstriyel makinalar, tıbbi ekipman), başarılı entegrasyon için geliştirme ekibinin genellikle donanım ürününüze erişim sağlaması gerektiğini unutmamalısınız. Bu ürünü bize ulaştırmanız gerekir. Ayrıca donanımınızı satan kişinin donanımla ilgili bilgisinin olması da önemlidir.

Mobil Uyumluluk

Eğer ürününüz mobil cihazlar için uyumlu değilse, piyasa sizden mobil uyumluluk sunmanızı bekleyecektir. Bunun için birkaç yöntem bulunmaktadır:

- Web tabanlı bir çözüm ise, ürününüzü mobil ekranlara uygun hale getirin ya destek alın.
- Yazılımınızın native veya cross-platform mobil uygulama versiyonunu oluşturun (uygulamayı hafif tutmak için yalnızca temel özellikleri dahil edebilirsiniz).

Yazılımınız belirli bir donanım için tasarlanmışsa, uzaktan kontrol için bu yazılım ürününü tamamlayıcı bir uygulama geliştirin.

Ürün Geliştirme Sürecini İş Perspektifinden Düzenleme

Ürününüzü kendi ekibinizle mi yoksa dış kaynak kullanarak mı geliştireceğinize karar verseniz bile, geliştirme sürecini iş perspektifinden düzenlemeniz gerekecek. Önemli noktaları gözden geçirelim.

Değer Odaklı Tasarım

'Değer' kavrayışınıza dayanarak, kullanıcı ilgisini, rekabet gücünü veya marka imajınızı ürününüzün özellik seçimini tanımlayan ve nihai hedef olarak belirleyebilirsiniz. Bu odaklı yaklaşım sayesinde ikincil özellikleri eler ve ürün geliştirme süresini ve maliyetlerini azaltabilirsiniz.

Bileşenlerin Yeniden Kullanımı

Parça parça her şeyi baştan yaratmak her zaman gerekli değildir. Ürün geliştirmeye başlamadan önce zaten mevcut olan çerçeveler, platformlar veya hizmetlerin kullanılabilirliğini göz önünde bulundurmak önemlidir. Mümkün olduğunca çok sayıda bileşeni tekrar kullanarak hem maliyetleri düşürebilir hem de geliştirme süresini kısaltabilirsiniz.

Versiyonlama

Ürününüzü ilk kez piyasaya sürmek için hazırlarken, ileriye dönük bir veya iki versiyon düşünmek önemlidir. Her iterasyonda kullanıcı geri bildirimlerine dayalı düzeltmeler yapılacak olsa da, sürümlemenizi sadece geçmiş hatalarınızla sınırlı tutarak ürünün gerçekten gelişmesini engellememelisiniz. Geri bildirimleri kullanarak hataları düzeltmek önemli olsa da, sadece bu düzeltmelerle yetinmek yerine ürünü sürekli olarak iyileştirmeye odaklanmalısınız.

Risk Yönetimi

Farklı risk alanlarını dikkate alın - zaman, bütçe, performans vb. - ve her birini özenle değerlendirin. Riskleri önceliklendirerek, olasılık ve olası zararlarına göre hareket ederek her biri için özel bir strateji geliştirmeye hazır olacaksınız. Risk yönetiminin önemli bir kısmı ürününüzün kontrolünü elinizde tutmakla ilgilidir - yani, başlangıç koduna, testlere, yapılandırma dosyalarına ve tüm gerekli belgelere erişim hakkınız olmasıdır. Bu sayede, sorunlu bir tedarikçi ile ilişkinizi sonlandırabilir ve farklı bir yazılım ürünü geliştirme şirketi ile işbirliğinizi sürdürebilirsiniz.

Önemli Notlar

- Fonksiyonel paketi seçmeden önce hedef kitlenizi iyi tanıyın ve gelecekteki kullanıcı senaryolarını belirleyin.
- Yayın sonrası sorunlarla karşılaşmamak için kalite ve güvenliğe odaklanın.
- Projenizin türüne göre, SaaS ve donanım entegrasyonunu düşünün; ayrıca uygulama performans izleme ve mobil erişimi önemseyin.
- Zaman, bütçe, performans ve tedarikçi risklerini kontrol altında tutun ve bunlara karşı hazırlıklı olun.
- Ürününüzün gelecek versiyonlarını düşünmeye başlayın ve gelişim planına bağlı kalırken geçmiş hataları düzeltin.