**GELENEKSEL YAZILIM SÜREÇ MODELLERİ:**

**Gelişigüzel Model:**

Aslında bu yöntemi model olarak isimlendirmemiz doğru olmaz. Kişisel bir yöntemdir denilebilir. Gelişigüzel geliştirme aşaması için herhangi bir model veya takip edilecek yol yoktur. Kişiye özel değişebilir yöntem ve metotların olduğu bir yol izlenir. Tek kişilik üretim ortamlarında görülen, basit programlamalar için kullanılan bir yoldur. Programı yazan kişi bile ileri zamanlar içerisinde geriye dönüp baktığında ne yaptığını anlayamayabilir. Bu nedenle ürünün bakım yapılabilirliği neredeyse imkansızdır. 1960 ‘lı Yıllarda basit programlamalar için kullanılmıştır.

**Çağlayan Yaşam-Döngü Modeli (Şelale-Waterfall):**

1970'li Yıllarda ortaya çıkan bir modeldir. Üretim sürecindeki temel adımları baştan sona en az bir kez izleyerek gerçekleşmektedir. Gereksinimleri iyi tanımlanmış ve kısa vadede üretilebilecek olan projeler için oldukça uygun bir modeldir. Barok modelinin aksine belgelemenin bir süreç aşamasından ziyade üretimin her aşamasında bulunup yapılması gerektiğini öngörür. Her aşama bir önceki aşama tamamlanmadan başlayamaz. Eğer ürün gereksinimleri iyi tanımlanmamış veya üretim süresi uzun sürecekse model işe yaramaz. Barok modelinin geri dönüş belirsizlikleri burada yoktur.

Genelde yazılımın kullanıcıya ulaşma zamanı uzun olduğu için bu modelde sorunlar yaşanabilmektedir.

Çoğu zaman gereksinim tanımlamaları kesin bir şekilde netleştirilemez. Bu kesinliğin olmaması nedeniyle hata ancak gerçekleştirme aşamasına gelindiğinde anlaşılabilir ve bu da epey zaman kaybına ve maliyete yol açar. Aynı zamanda bu hatalardan kaynaklı zaman kayıpları iş ortamında çalışanların motivasyonunu düşürürken müşteriye karşı da firmanız adına kötü bir imaj çizer.

**V Süreç Modeli:**

Bu modeli çağlayan modelin uyarlanması olarak düşünebiliriz. Model V harfini andırırken sol tarafı üretimi, sağ tarafı ise sınama işlemleri ile ilgili aşamalardan oluşmaktadır. Her bir üretim aşamasının karşısında bulunan sınama aşamaları sayesinde hata kaynaklarına dönüş daha kolaydır.

Modelde temel çıktılar olarak: kullanıcı modeli, mimari model ve gerçekleştirim model olarak isimlendirilmiş üç alt model vardır.

Yine çağlayan modelde de olduğu gibi belirsizliklerin az ve gereksinim tanımlamalarının net olduğu projelerde V süreç modeli uygun bir model olarak ele alınır.

**Helezonik (Spiral) Model:**

Risk analizi konusunun özenle üzerinde duran bir modeldir. Ürettiği Prototipler sayesinde yinelemeli artırımlı bir yaklaşımı vardır.

Planlama, risk analizi, üretim ve kullanıcı değerlendirmesi adında 4 ana aşamadan oluşup tekrarlanır. Tekrarlanırken yaptığı her bir tekrara faz denir. Bu fazlar esnasında risk analizlerinin ardından kullanıcıdan olumlu ya da olumsuz yanıt almak için, ürünün prototipi üretilip kullanıcıya sunulur. Her sunumun ardından alınan değerlendirmeler sonucu yeni geliştirilmiş planlarla yeni bir faza başlanır.

Modelin en önemli avantajlarından bir tanesi kullanıcı katkısının üretimde önemli bir pay sahibi olmasıdır. Bu katkı sayesinde ürün kullanıcının isteklerinin tamamına yanıt verebilirken, üretim esnasında da hatalar oldukça erken fark edilerek zaman ve maliyet kaybı önlenmiş olur.

Helezonik model komplike bir yapıya sahiptir. Spiral bazen sonsuza dek sürebilir. Bu süreçte belgelemeler oldukça fazla olacak ve karmaşıklık daha da artacaktır. Küçük ve düşük riskli projeler için pahalı bir modeldir.

**Çevik (AGILE) Yazılım**

Gelişen teknolojilerle büyüyen yazılım dünyasında, kullanıcıların daha kaliteli ürünlere daha ucuz ve daha kısa sürede ulaşmak istemesi, yazılım sektöründe çevik yöntemlerin gelişmesine yol açmıştır. Bu yöntemlerin amacı, geliştirme sürecindeki bazı yüklerden kurtularak sonuca daha hızlı ulaşıp, değişen ihtiyaçlara daha erken cevap vermektir. Aynı zamanda yüksek performans sağlayan, hata riski en az ve ucuz yollu çözümler oluşturmak da olduğunu söyleyebiliriz.

Çevik geliştirme metodu; değişimi, parça parça yazılım teslimatını, birlikte uyumlu çalışmayı, test odaklı yazılım geliştirilmesini ve uyumlu planlamayı teşvik eder. Birlikte çalışmayı teşvik eden bu metot yazılımcı arkadaşların da morallerini yüksek tutmayı sağladığı için üretim daha kolay ve hızlı bir şekilde devam ederken sonuca gelindiğinde kaliteli ürünler elde edilir.

Kanban ve Scrum en popüler iki yöntemdir.

**Scrum’ın Popülerliği:**

Karmaşık yapıları basite indirgeyerek, ürün sahibi ile yapılan toplantıların ardından gereksinimleri anlaşılır bir hale getiren, ürün geliştirme metodu scrum ekip çalışmasına da önem verir. Ekip arasındaki iletişimi ve motivasyonu yukarı çeken bir metot olması da şirketler için artılarından bir tanesidir. Günlük yapılan kısa toplantılar ile de kontrolleri sağlayan metot hata ve takım arasındaki iletişim kopukluklarına da engel olur. Bu yüzden scrum metodu yazılım geliştiriciler açısından oldukça popüler ve kullanışlıdır.

**Scrum nedir? Scrum Teorisi neler anlatır?**

Scrum, agile ekipleri arasında en yaygın kullanılan yapıdır. Agile, karmaşık işleri / projeleri yönetmek ve organize etmek için bazı temel değerler ve kurallar tanımlar. Bir düşünce tarzı olarak bahsedebileceğimiz agile karşısında scrum, biraz daha katı kuralları olan bir proje yönetim prensibidir diyebiliriz.

Scrum teorisi 3 temel değere dayanır;

* Şeffaflık : Ekip, scrum boyunca uğruna çalıştığı ürünün mevcut durumunu görebilmeli.
* Denetleme : Herkesi aynı noktada tutabilmenin tek yolu sık sık kontrol noktalarında soluklanmaktır.
* Adaptasyon : Ekipten biri dahi sprint planından saptığında planda en hızlı şekilde yeniden düzenlemeler yapılmalıdır.

Scrum kendi içinde sık sık tekrar eden ve bunu bir öncekinden hareketle her seferinde daha iyi yapan bir süreçtir. Her tekrar, kendi içinde iyileştirecek bir şeyler bularak ve uygulayarak bir sonrakine geçişi hedefler. Geliştirici ekipler, müşterinin talep ettiği özellikleri ortaya çıkarmak için küçük planları doğru bir şekilde tamamlayarak ilerler.

**Scrum uygulamanın faydaları nelerdir?**

Ekip içi iletişimi iyileştirmek, ekibin birlikte çalışmasını kolaylaştırmak, ortaya daha verimli sonuçların çıkmasını sağlamak ve odaklanmayı arttırmak için pek çok [ekip lideri](https://www.netuce.com/yazilim-ekibi-liderleri-icin-hayat-kurtaracak-10-oneri/) scrum uygulama yoluna gider. Scrum uygulamak ekiplere esneklik sağlar. Bu sayede, mükemmel hazırlanmış planlar paralelinde gerçek dünyanın getirdiği zorunlu değişimlere çok daha hızlı adapte olup en yüksek verimlilikle çalışmaya devam etmek mümkün olur.

Scrum kullanan takımlar, projenin içerdiği tüm adımların herkes için eşit derecede görünür olmasını sağlar. Bu sayede bilgi herkese eşit dağılır ve herkes aynı noktada düşünebilir. Genellikle satış sorumlusu, müşteri temsilcisi gibi işin işletme tarafında olan kişiler ile teknik ekipler bambaşka diller konuşur. Bu sebepten doğan bilgi eksiklikleri iletişimin oldukça zorlayıcı bir hale gelmesine sebep olabilir. Scrum tüm bu proje ortaklarının aynı dilden konuşabilmesini sağlar.

Scrum uygulamakla beraber ekibin sahip olduğu verimlilik arttıkça, projelerle ilgili finansal geri dönüşler de benzer şekilde artışa geçer. Ürünlerinizi belirlediğiniz tarihlerde pazara sunabilir ya da tam zamanında teslim ettiğiniz işlerle müşterinizi mutlu ederek onların da sizi mutlu etmesini sağlayabilirsiniz. Çünkü scrum ile eskiye oranda daha karmaşık süreçlerin içinden daha kolay çıkabilir, geliştirme süreçlerini hızlandırabilirsiniz. Bunun için scrum değerlerini ekipçe benimseyip temel scrum prensiplerinden sapmamakta fayda var.

**Sprint nedir?**

Çoğunlukla 2 ila 4 haftalık kısa planlamalardır. Geliştirici ekip, her sprintin sonunca ürüne bir özelliği daha eklemiş olmayı hedefler. Bu nedenle özelliğin kapsamına göre sprint süreleri değişkenlik gösterebilir. Sprint sonlarında başarılı bir demo yapabilmeyi hedeflemek iyi bir motivasyon kaynağıdır. Bu aynı zamanda tüm proje ortaklarını ürünle ilgili aynı bilgi seviyesinde tutmaya yardımcı olur. Sprintler, özelliği ortaya çıkarmak için parçalanmış görevlerden oluşur. Ekipte herkes bu görevlerden birini üstlenerek sprint boyunca çalışır. Ancak sprinti tamamlamak her zaman ekibin sorumluluğundadır. Bu nedenle de sprint boyunca görev dağılımları ihtiyaca göre şekilde değiştirebilir.

**Scrum ekibinde yer alan roller nelerdir?**

[Scrum guide](https://scrumguides.org/scrum-guide.html)’a göre bir scrum ekibinde mutlaka bulunması gereken 3 temel rol şunlardır;

1. Product owner
2. Scrum master
3. Development Team

Scrum ekibi, proje amaçlarını yerine getirmek üzere tüm scrum toplantılarında beraber çalışır.

**PRODUCT OWNER**

Product owner, ekibin üzerinde çalıştığı projenin vizyoneridir. Çalışma sonunda ortaya nasıl bir ürün çıkacağına, bu ürünün hangi özelliklere sahip olacağına, ürünün nasıl görüneceğine ve neye benzeyeceğine product owner karar verir. Aslında ekibin öncüsüdür. Product owner ürünün değerini belirler ve ürün backlogunu (product backlog) yönetir.

**Product backlog nedir?**

Geliştirici ekibin tamamlaması gereken tüm işler burada birikir. Burası ürünün ihtiyacı olan ve ileride olacak tüm tespitlerin yer aldığı yerdir. Bu nedenle burası hiçbir zaman tamamlanmaz ve sürekli olarak gelişmeye ve çoğalmaya devam eder.

**Product owner sorumlulukları nelerdir?**

* Product backlogu yönetir; backlog öğelerini tanımlar.
* Product backlog için bir hedef belirler ve buna göre backlog ayarlamaları yapar.
* Scrum ekibindeki diğer takım arkadaşlarının ortaya çıkardığı işin değerinin sürekli artmasını sağlar.
* Backlogtaki görevleri scrum takımları için her zaman açık ve anlaşılır olmasını sağlar.
* Tüm bunlarla beraber product owner sprinti iptal etmeye yetkisine de sahiptir. Eğer sprint başlangıcındaki hedef artık takım için geçerli değilse, product owner sprinti iptal edebilir. Ancak bunun zaman, efor ve para maliyetinin sorumlumluluğunu da kendisi üstleniyor olmalıdır.

**İyi bir product owner olmak için neler gerekir?**

* İyi bir product owner, İyi bir karar vericidir. Her zaman son kararı o verir.
* Aldığı kararları açık ve doğru bir şekilde ekibe ifade edebilir.
* Ürün ve ürünün içerisinde bulunduğu pazar hakkında bilgi sahibidir.
* Hem ekibin hem de müşterinin güvenine sahip olmalıdır bu nedenle iyi bir ikna edicidir.
* Müşteriyi iyi anlar ve müşteriden aldıklarını ekibe aktarmada iyi bir çevirmendir.
* Ürün için en doğru özellikleri seçip kalanlara hayır diyebilen ve bu kararına sadık kalandır.
* Ürün için geliştirme saatlerinin belirlendiği scrum toplantılarına ve sprint sonlarında sprint değerlendirmesi yapılan retrospective toplantılarına katılır.
* Bir scrum ekibi için o andaki en ideal süreci product owner belirler.

**SCRUM MASTER**

Scrum master, herkesi bir arada tutan ve işlerin olması gerektiği şekilde, zamanında teslim edilmesini sağlayan kişidir. Ekibi dikkat dağıtıcı etkenlerden korur, ekibin gelişmesine ve ilerlemesine destek olur, ekibe scruma karşı sorumluluklarını sık sık hatırlatır.

**Scrum değerleri**

Scrum değerlerinin korunması, bir scrum master için en önemli sorumluluklardan biridir. Agile metodolojisinde tanımlanmış 5 scrum değeri vardır;

* Cesaret
* Odaklanma
* Bağlılık
* Saygı
* Açıklık

Scrum master için bilinen en yaygın görev, daily scrum toplantılarını yapmaktır; ancak bilinenin aksine asıl sorumluluğu takımın daily scrum toplantılarını yaptığından emin olmaktır. Scrum master takıma koçluk eder ve işlerin yolunda gitmesini sağlamaya çalışır.

**Bir scrum master için günlük koşuşturmada neler vardır?**

Scrum master için pek çok tanımlanamayan sorumluluk sayabiliriz. Çünkü her scrum takımının kendine ve çalıştığı projeye özgü ihtiyaçları olabilir. Genellemek gerekirse scrum master, takımı bir arada tutan ve ona hizmet eden bir roldür. Bu anlamda scrum masterın günlük işlerine baktığımızda aşağıdaki kilit işleri sayabiliriz:

* Geliştirici takımın sprintteki tüm görevleri anladığından emin olur.
* Geliştirici takıma koçluk eder; geliştirici ekip herhangi bir ihtiyaç durumunda ilk olarak scrum master ile görüşür.
* Aynı zamanda mental olarak da koçluk yapar. Takımın kendine inanmasını ve performansının en üst seviyesinde olmasını sağlar.
* Takımın proje ya da bireysel ihtiyaçları ile ilgili scrum masterdan koçluk talep etmelerinin tek yolu, güçlü liderlik özelliklerine sahip olmaktır. Takım scrum masterı saygı duyduğu ve akıl almak isteyeceği biri olarak görmeli.
* Daha iyi bir backlog yönetimi için araştırma yapar, yeni teknikler üretir.
* Scrum seramonilerini ekip için sürekli olarak daha kolay uygulamanın yollarını arar.
* Geliştirici ekibin herseferinde daha iyi bir çıktı elde etmesi için, ekibe hizmet eder.
* Scrum takımının scrum değerleri ve çerçevesinde çalışmaya devam etmesini sağlar.
* Scrum master elindeki tüm imkanları takım için harcar. Her zaman önce takım sonra kendisi gelir.

**GELİŞTİRME EKİBİ**

**Development ekip scrum takımlarının olmazsa olmaz üyeleridir. Developer ekip, sprintler boyunca bir ürün ya da ürün özelliği için product owner tarafından belirlenmiş işleri, onları nasıl sonuçlandıracağına kendisi karar vererek işleri üstlenir ve bunu sprint sonuna kadar gerçekleştirir.**

Ekibi oluşturan developerlar ve testerların en önemli hedefi, koştuğu sprinti zamanında ve istenen sonucu üretecek şekilde tamamlamaktır. Bunu ancak bir ekip olarak davranabildiğinde gerçekleştirebilir. Sprint boyunca kararları bireysel değil; bir ekip olarak verir. Sprint boyunca günlük aktiviteler üzerine yoğunlaşır ve her geçen gün sprintteki görevleri eritir. Scrum ekiplerinin tamamı için süreci anlaşılır ve şeffaf kılmakla yükümlüdürler. Yani geliştirme ekibi sprint boyunca da yapılan işin hangi seviyede, ne durumda olduğunu anlaşılır bir şekilde ifade edebilmelidir.

Geliştirme takımları, genellikle çok kalabalık ekipler değildir. En az 3 en fazla 9-10 kişilik ekipler ve her ekibin kendi scrum master ve prodcut owner’ı olur. Daha küçük takımlar ile sprint tamamlamayı ve daha kalabalık takımlar için kaliteli bir iletişim kurmayı beklemek çok gerçekçi olmayacaktır.

**İyi bir geliştirme ekibinin özellikleri nelerdir?**

* **Takım zihniyetine sahiptir. Takımdaki herkes birbirine ve yeteneklerine güvenir. Takım beraber iyi çalışır.**
* **Çapraz fonksiyoneliteye sahiptir. Alınması beklenen tüm sonuçlar için ihtiyacı olan becerilerin tamamına sahiptir.**
* **Takımdaki herkes eşittir. Takım içerisinde hiyerarşi yoktur ve herkesin eşit derecede sorumluluğu vardır.**
* **İyi bir geliştirme takımı kendi kendine organize olabilir. Herhangi bir scrum master ya da başka birinin işleri nasıl yapacaklarını söylemesine veya onları yönetmesine ihtiyaç uymaz.**

**Scrum Toplantıları**

**Sprint Planlama (Sprint Planning)**

Sprint, planlama toplantısı ile başlar. Scrum master, geliştirme ekibi ve ürün sahibinin katılımı ile bu toplantı gerçekleştirilmektedir. Planlama toplantılarının amacı, ürün için talep edilen geliştirmelerin yer aldığı *ürün iş listesi* (product backlog) içerisindeki işlerden bir kısmının geliştirme ekibi ile paylaşılmasıdır. Bu toplantının sonunda aktif sprint içerisinde yapılmasına karar verilen işler, ürün iş listesinden *sprint iş listesine* (sprint backlog) alınır.

**Günlük Scrum Toplantısı (Daily Scrum Meeting)**

Geliştirme takımından herkesin katılımı ile günlük toplantılar yapılmaktadır. Scrum master ünvanındaki kişinin de toplantıya katılması ve günlük toplantı kurallarına uyulduğundan emin olması önerilir. Ürün sahibinin katılması zorunlu olmasa da katılım göstererek süreç hakkında bilgi alması için iyi bir fırsattır.

Günlük toplantılarda geliştirme takımında yer alan kişiler *dün neler yaptıklarını*, *bugün neler yapacaklarını*anlatırlar. Bunları anlatmalarının amacı hesap vermek değil, takım arkadaşları ile bilgi paylaşmaktır. Üstlenilen görevlerin tamamlanması konusunda bloklanılan, yardım beklenen bir konu varsa bu toplantıda belirtilir.

Zaman sınırı olan bir toplantıdır. 15 dakika gibi bir sürede toplantının tamamlanması beklenir. Toplantının uzamasına mahal vermemek için bu toplantılar genelde ayakta yapılmaktadır.

**Sprint Değerlendirme (Sprint Review)**

Scrum takımı üyeleri (scrum master, ürün sahibi ve geliştirme takımı) ve önemli paydaşların katılımı ile bu toplantı yapılmaktadır. Bu toplantının amacı sprint süresince geliştirmesi tamamlanmış ve ürünün bir parçası haline gelmiş özelliklerin tanıtılmasıdır. Bir nevi ürünün reklamını yapmak da diyebiliriz. Paydaşlardan geri bildirimler alınır ve gerekiyorsa ürün üzerinde revizyonlar yapılır. Paydaşların ürün gelişim sürecine dahil olmaları için kullanılan kritik toplantılardan biridir.

**Sprint Retrospektif (Sprint Retrospective)**

Scrum takımının tamamının katılımı ile yapılan toplantıdır. Sprint sürecinin takım olarak değelendirilmesi amacıyla düzenlenir. *Nelerin daha iyi yapılabileceği* veya *nelerin iyi yapıldığı* konuları üzerine konuşulur. Ekip arkadaşları birbirlerine geri bildirimler verir ve bu sayede daha iyi işleyen bir ekibin oluşması amaçlanır.

İyi (Glad), Kötü (Sad), Sinir Bozucu (Mad) veya Başlayalım (Start), Devam Edelim (Continue), Yapmayalım (Stop) gibi farklı formatları olan bir toplantıdır. Açık iletişim ve üyeler arasında güven bu toplantının olmazsa olmazları arasındadır.

**KANBAN**

Yalın ürün geliştirme, üretim süreçlerinde başarıyla kullanılan yalın metodolojilere dayanmaktadır. **Kanban** bir **agile proje yönetim** tekniğidir.

Kanban, bir süreç içinde hareket ederken işi yönetmek için kullanılan görsel bir sistemdir.

**Kanban**, hem süreci (iş akışını) hem de bu süreçten geçen fiili işi görselleştirir.

**Kanban**'ın amacı, sürecinizdeki olası darboğazları belirlemek ve bunları düzeltmektir, böylece iş, optimum hızda veya verimde gerçekleşebilir.

**Kanban** iş akışını optimize etmenize ve ekibinizin tam kapasitesini kullanmanıza yardımcı olmak için tasarlanmış bir yöntemdir.

**Kanban İlkeleri**

Kanban, ekip yapınızda herhangi bir değişiklik yapmadan süreçleri ve iş akışı verimliliğini iyileştirmek için kullanılabilir.

**Kanban Sistemi’nin İlkeleri**

**1) Şu anda yaptığınız şeyle başlayın:** Kanban, mevcut iş akışların, sistemlerin ve süreçlerin üzerinde, halihazırda olanı bozmadan yöntemi kullanma esnekliği sunmaktadır. Yöntem, mevcut süreçlerin, rollerin, sorumlulukların ve unvanların değeri olduğunu ve korunmasını gerektiğini kabul etmektedir. Herhangi bir değişikliğe ihtiyaç duyulması halinde süreç, ekibin rahat edebileceği bir hızda belirli bir süre içinde kademeli olarak gerçekleştirilmektedir.

**2) Giderek Artan, Evrimsel Değişimi Takip Etmeyi Kabul Edin**: Kanban yöntemi, minimum direnci karşılayacak şekilde tasarlanmıştır. İş birliği ve geri bildirim formları uygulayarak mevcut süreçte sürekli giderek artan evrimsel değişiklikleri teşvik eder.

**3) Başlangıçta mevcut rollere, sorumluluklara ve iş unvanlarına saygı gösterin:** Kanban, diğer yöntemlerden farklı olarak kendi başına herhangi bir organizasyon değişikliği dayatmaz. Bu nedenle, iyi performans gösterebilecek mevcut rol ve işlevlerinizde değişiklik yapmanıza gerek yoktur. Ekip, gerekli değişiklikleri iş birliği içinde belirleyecek ve uygulayacaktır.

**4) Tüm seviyelerde liderlik eylemlerini teşvik edin:** Kanban, organizasyonun tüm seviyelerinde sürekli iyileştirmeyi teşvik eder ve liderlik eylemlerinin yalnızca üst düzey yöneticilerden kaynaklanmak zorunda olmadığını söyler. Her seviyedeki insanlar, ürün ve hizmetlerini sunma biçimlerini sürekli iyileştirmek, değişiklikleri uygulamak için fikirler sağlayabilir ve liderlik gösterebilir.

**6 Adımda Kanban Metodu Uygulaması**

·        **1. Süreci görselleştirmek:** Kanban Metodunu benimsemenin ve uygulamanın temel ilk adımıdır. İşinizi veya hizmetlerinizi sunmak için şu anda kullandığınız işlem adımlarını fiziksel bir panoda veya elektronik bir Kanban Panosu üzerinde görselleştirmeniz gerekir. Sürecinizin karmaşıklığına ve iş karışımınıza bağlı olarak, Kanban panonuz çok basit ila çok ayrıntılı olabilir. Sürecinizi görselleştirdikten sonra, sizin ve ekibinizin yaptığı mevcut işi görselleştirebilirsiniz.

·        **2. Devam Eden Çalışmayı Sınırlamak:** Devam eden çalışmayı sınırlamak, “Çekme sistemi" olan Kanban'ı uygulamak için esastır. Devam Eden Çalışmayı sınırlayarak, ekibinizi yeni bir işe başlamadan önce eldeki işi tamamlamaya teşvik etmiş olursunuz. Bu nedenle, şu anda devam etmekte olan iş tamamlanmalı ve tamamlandı olarak işaretlenmelidir. Bu, sistemde kapasite yaratır, böylece ekip tarafından yeni işler çekilebilir.

·        **3. Akışa Odaklanmak:** Akışı yönetmek ve iyileştirmek, ilk 2 uygulamayı uyguladıktan sonra Kanban sisteminizin en önemli noktasıdır. [Kanban sistemi](https://unopro.com.tr/hizmetler/egitim/dijital-model-fabrika-uygulamali-yalin-ve-dijital-donusum-workshoplari/heijunka-seviyelendirme-kanban), iş akışının çeşitli aşamalarını ve her aşamadaki işin durumunu vurgulayarak akışı yönetmenize yardımcı olur. İş akışının ne kadar iyi tanımlandığına ve Devam Eden Çalışma Limitlerinin ne kadar iyi ayarlandığına bağlı olarak, Devam Eden Çalışma sınırları dahilinde düzgün bir akış gözlemleyecek veya bir şey takılıp kapasiteyi artırmaya başladığında işlerin biriktiğini gözlemleyeceksiniz.

·        **4. Süreç Politikalarını Açık Hale Getirmek:** Sürecinizi görselleştirmenin bir parçası olarak, yaptığınız işi nasıl yaptığınıza ilişkin politikalarınızı (süreç kuralları veya yönergeleri) açıkça tanımlamanız ve görselleştirmeniz de gerekmektedir. Açık süreç yönergelerini formüle ederek, tüm katılımcıların sistemdeki herhangi bir çalışmayı nasıl yapacaklarını anlamaları için ortak bir temel oluşturursunuz. Politikalar pano düzeyinde olabilir. Her bir iş öğesi türü, her bir sütun için giriş-çıkış kriterleri veya ekip üyelerinin panodaki iş akışını iyi yönetmesine yardımcı olan herhangi bir şey için yapılması gereken adımların bir kontrol listesi olabilirler. Politikalar açıkça tanımlanmalı ve genellikle panonun üstünde ve her şeritte ve sütunda görselleştirilmelidir.

·        **5. Geri Bildirim Döngülerini Uygulamak:** Geri bildirim döngüleri, herhangi iyi bir sistemin ayrılmaz parçasıdır. Kanban Metodu, çeşitli türlerde geri bildirim döngülerini uygulamanıza yardımcı olur. Kanban panosu iş akışınızdaki gözden geçirme aşamaları, ölçümler ve raporlar ve sisteminizdeki işin ilerleyişi veya eksikliği hakkında size sürekli geri bildirim sağlayan bir dizi görsel ipucu olacaktır.

·        **6. İş birliği İçinde Geliştirin:** Kanban Yöntemi, evrimsel bir iyileştirme sürecidir. Küçük değişiklikleri benimsemenize ve ekibinizin kolayca üstesinden gelebileceği bir hızda ve boyutta kademeli olarak gelişmenize yardımcı olur. Bilimsel yöntemin kullanılmasını teşvik eder. Bir hipotez oluşturursunuz, onu test edersiniz ve testinizin sonucuna göre değişiklikler yapabilirsiniz.