

Architekturkicker



Swing Frontend

Es gibt bereits eine Schwester-Applikation die von den gleichen Benutzern bedient wird. Diese verfügt über ein Java Swing-UI. Die Benutzer sollen ohne „Bruch“ mit den beiden Applikationen arbeiten können.



www.embarc.de

Architekturkicker



Integration

Es gibt einen Unternehmensstandard für die Kommunikation von Systemen untereinander. Die entwickelte Anwendung muss Daten über einen ESB und ein definiertes XML-Schema entgegennehmen und wieder bereit stellen.



www.embarc.de

Architekturkicker



Legacy

Aus dem Portfolio an Altanwendungen ist eine besonders spannend - Sie bezieht sich teilweise auf den selben Datenstamm und wird auch weiterhin verwendet.



www.embarc.de

Architekturkicker



Fremder Betrieb

Die Anwendung wird mehr als einmal und in unterschiedliche Hardware/Server Konfigurationen ausgeliefert. Die Ziellandschaften und die Fähigkeiten des Betriebspersonals sind nur geringfügig steuerbar.



www.embarc.de

Architekturkicker



Eigenes Hosting

Im Firmeneigenen Rechenzentrum werden viele Host-Applikationen und JEE-Anwendungen betrieben. Die neue Anwendung soll hier keine Ausnahme bilden.



www.embarc.de

Architekturkicker



.NET Know How

Das Entwicklungsteam kennt sich hauptsächlich mit Standard .NET Anwendungen aus. Entwicklungstools, Build und Deployment sind auf dieser Basis gut verstanden.



www.embarc.de

Architekturkicker



Keine Cloud-Lösung

Das Unternehmen traut den momentanen Cloud-Plattformen nicht. Public Cloud wird somit als Infrastruktur, Plattform und Service abgelehnt.



www.embarc.de

Architekturkicker



Aufbewahrungspflichten

Aufgrund rechtlicher Rahmenbedingungen müssen einige Daten der Applikation mindestens 10 Jahre aufbewahrt werden und dürfen nicht verfälscht werden.



www.embarc.de

Architekturkicker



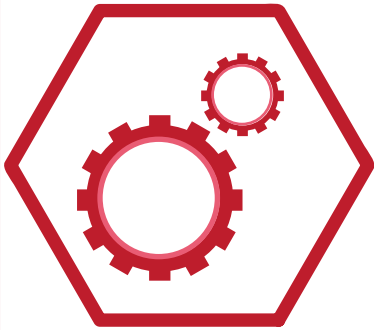
Kein Open-Source

Aufgrund Unternehmensinterner Richtlinien ist der Einsatz von Open-Source Bibliotheken und Frameworks untersagt.



www.embarc.de

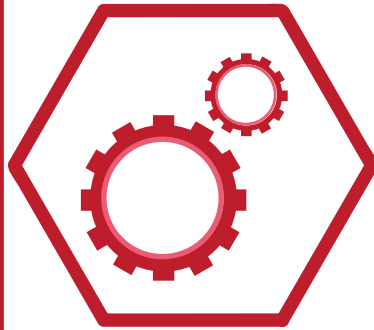
Architekturkicker



www.embarc.de



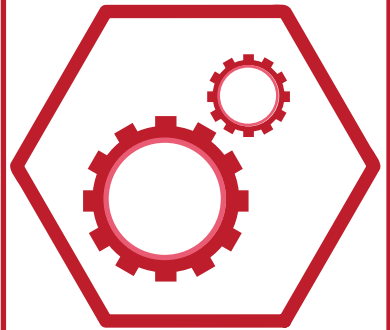
Architekturkicker



www.embarc.de



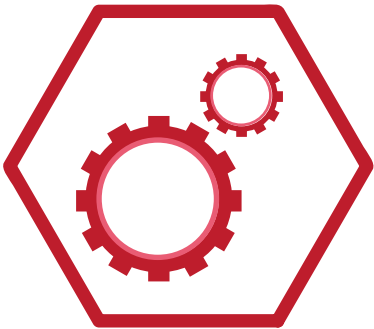
Architekturkicker



www.embarc.de



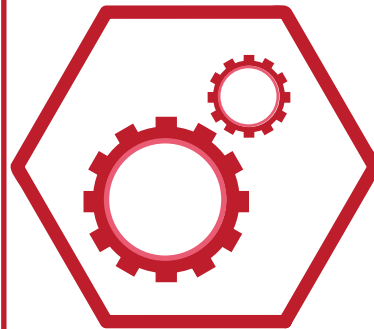
Architekturkicker



www.embarc.de



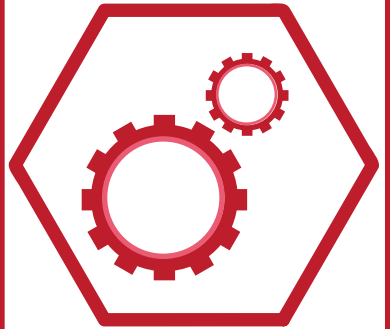
Architekturkicker



www.embarc.de



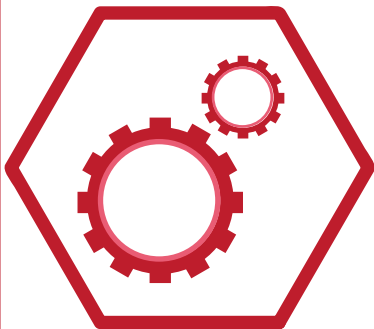
Architekturkicker



www.embarc.de



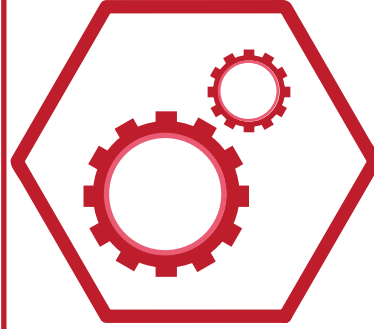
Architekturkicker



www.embarc.de



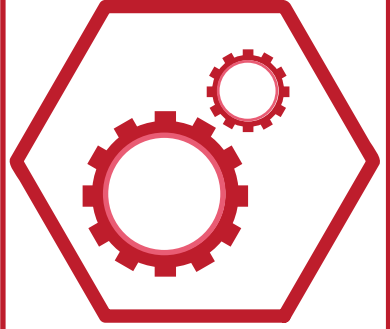
Architekturkicker



www.embarc.de



Architekturkicker



www.embarc.de



Architekturkicker



Process Engine

Aus politischen Gründen (angeschaffte Lizenzen, Abteilungsnähe) ist der Einsatz einer Business-Process-Engine (BPE) zur Steuerung der Geschäftsprozesse gesetzt.



www.embarc.de

Architekturkicker



Oracle Datenbank

Es gibt einen bestehenden Support-Vertrag für Oracle Datenbanken. Sie bilden die Datensinke für alle Applikationen der Organisation. So auch für die entwickelte Anwendung.



www.embarc.de

Architekturkicker



2-Phase-Commit

Die Speicherung von Daten mit Relevanz für andere Systeme erfordert nach Unternehmensstandard ein 2-phase-commit, also den Einsatz eines Transaktionsmonitors der im Falle von Fehlern in abhängigen Systemen die Datenänderung im eigenen System zurückrollt.



www.embarc.de

Architekturkicker



CQRS

Die Anwendung soll unter anderem dazu dienen Erfahrung mit dem Command-Query-Responsibility Segregation-Pattern zu machen, in dem das Datenmodell für lesende Zugriffe von jenem für schreibende Zugriffe getrennt wird.



www.embarc.de

Architekturkicker



Cloud

“Weil die Zukunft nicht auf sich warten lässt” hat das Unternehmen beschlossen nicht weiter in einen eigenen Betrieb / in eigene Hardware zu investieren. Für das aktuelle Projekt ist eine Cloud-Lösung ins Auge zu fassen.



www.embarc.de

Architekturkicker



Komponentendefinition

Das Unternehmen verfügt über ein Set von Architekturregeln die zentral erarbeitet wurden und in jedem Projekt eingehalten werden müssen. Eine davon:
- Komponenten sind Pakete, die ausschließlich Abhängigkeiten zu eigenen Interfaces haben.



www.embarc.de

Architekturkicker



Datenmodell

Das Unternehmen verfügt über ein Set von Architekturregeln die zentral erarbeitet wurden und in jedem Projekt eingehalten werden müssen. Eine davon:
- Das Datenmodell wird für die Anwendung zentral entworfen und optimiert.



www.embarc.de

Unternehmenskultur



Ordnung muss sein!

Auf Korrektheit und Einheitlichkeit wird hier großer Wert gelegt und vom Chef bis zum Praktikanten gelten gleichermaßen feste, detaillierte Regeln und Strukturen. Die Verantwortung von Vorgesetzten besteht darin auf ihre Einhaltung zu achten. Fehlverhalten wird konsequent sanktioniert.

www.embarc.de

Unternehmenskultur

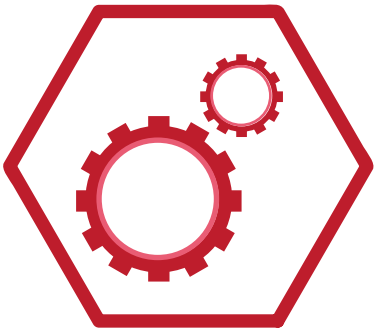


Der Erfolg gibt uns Recht!

Für die Erfolgs- und Gewinnmaximierung, wird hier jederzeit die Strategie auch wieder umgeschmissen. Wettbewerb, Leistung und Status stehen im Vordergrund, weshalb es immer auch vom Vorteil ist eine wenig andersartig und „individuell“ zu sein. Zumindest sollten man einen wichtigen Titel oder den größten Laptop haben.

www.embarc.de

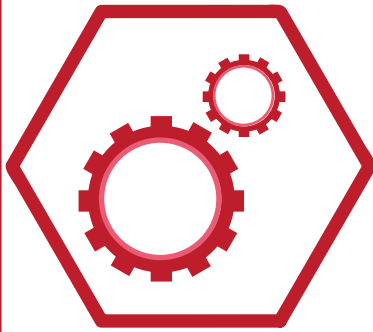
Architekturkicker



www.embarc.de



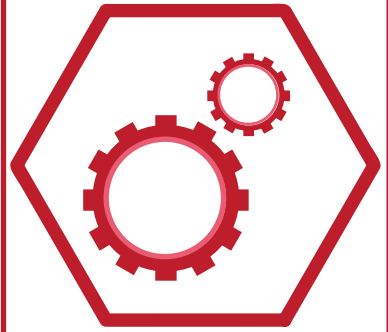
Architekturkicker



www.embarc.de



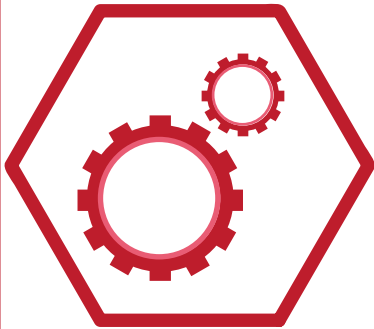
Architekturkicker



www.embarc.de



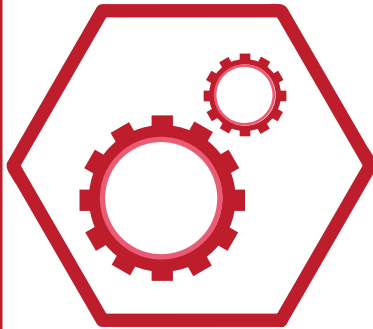
Architekturkicker



www.embarc.de



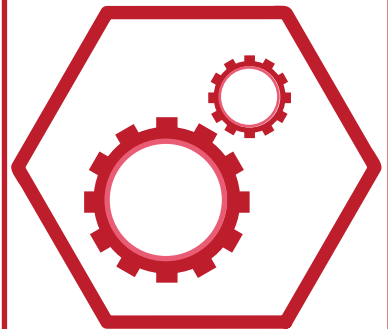
Architekturkicker



www.embarc.de



Architekturkicker



www.embarc.de



Unternehmenskultur



www.embarc.de



Unternehmenskultur



www.embarc.de



Architekturkicker



www.embarc.de



Methodikkicker



Klassischer Architekt

Es gibt einen definierten Architekten, der alle zentralen technischen Entscheidungen verantwortet. Er erstellt eine Blaupause an die sich die Entwickler weitestgehend halten müssen.



www.embarc.de

Methodikkicker



Gemeinsame Entscheidung

Betreffen Entscheidungen zu Technologien, Strukturierung, etc. mehrere Entwickler oder Teams, bereitet ein Entwickler eine Entscheidung vor. Der Vorschlag wird dann von allen Interessierten per Veto-Abfrage angenommen.



www.embarc.de

Methodikkicker



Prinzipien

Nur sehr grundlegende Architekturentscheidungen werden direkt getroffen. Alles unterhalb des groben Applikationsrahmens wird mit Prinzipien geregelt und lokal entschieden.



www.embarc.de

Methodikkicker



Architekturbewertung

Alle 6 Wochen trifft sich das Entwicklungsteam mit anderen Stakeholdern und evaluiert ob die momentane Ist-Architektur der Zielerreichung zuträglich ist, bzw. welche Risiken momentan bestehen.



www.embarc.de

Methodikkicker



Anforderungsgetriebene Architektur

Architekturarbeit wird durch konkrete Aussagen zu nicht-funktionalen Aspekten getrieben. Diese werden priorisiert und mit den funktionalen Anforderungen vermengt abgearbeitet.



www.embarc.de

Methodikkicker



Doku im Wiki

Die Architekturdokumentation besteht aus einem Dashboard im Wiki und verlinkten Seiten, die zentrale Entscheidungen und die generelle Architekturidee erklären.



www.embarc.de

Methodikkicker



ADD

Ein Architecture Definition Document beschreibt die Architektur ausführlich und detailliert. Die enthaltenen Sichten beschreiben die Softwarearchitektur aus mehreren Blickwinkeln und für unterschiedliche Stakeholder.



www.embarc.de

Unternehmenskultur



Uns geht es doch gut!

Damit es uns allen gut geht müssen wir auf alle achten und alle miteinander im guten Kontakt sein. Harte Ziele oder gar Hierarchien stören da, Mitsprache und Gleichberechtigung von allen bei allem sind wichtig. Und wenn der Kunde dabei mal aus dem Auge verloren wird, ist das nicht so wichtig wie das gemeinsame Wohlbefinden.

www.embarc.de

Unternehmenskultur

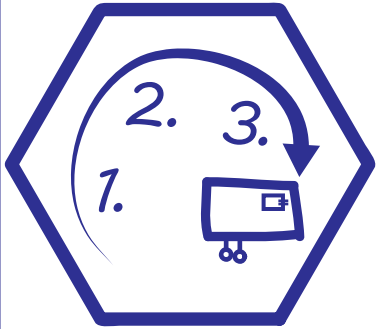


Das Beste setzt sich durch!

Bei der Suche nach der optimalen Lösung bedient man sich am Besten und bei den Besten die gerade da sind. Teams und Gruppen entstehen und zerfallen je nach Bedarf und Problemstellung und bilden ein Netzwerk. Führung erfolgt durch Kompetenzträger, Verantwortung wird situativ geteilt und übertragen.

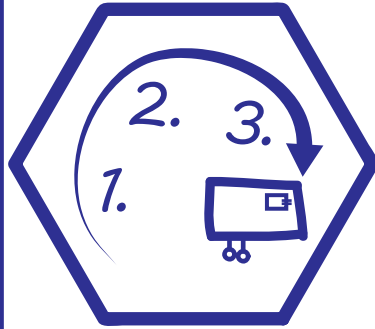
www.embarc.de

Methodikkicker



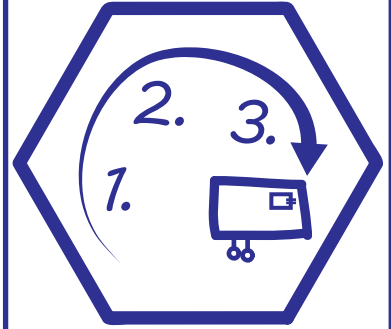
www.embarc.de 

Methodikkicker



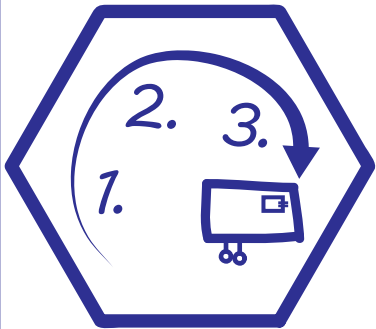
www.embarc.de 

Methodikkicker



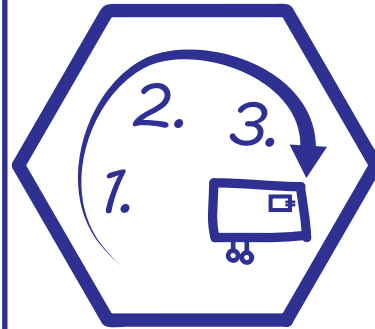
www.embarc.de 

Methodikkicker



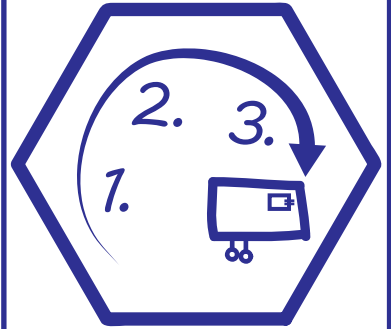
www.embarc.de 

Methodikkicker



www.embarc.de 

Methodikkicker



www.embarc.de 

Unternehmenskultur



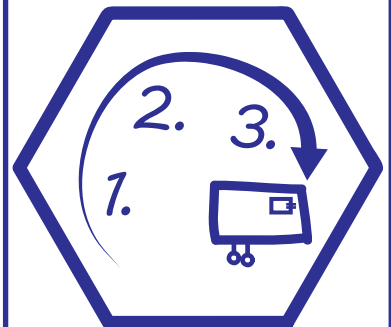
www.embarc.de 

Unternehmenskultur



www.embarc.de 

Methodikkicker



www.embarc.de 

Methodikkicker



UML-Modellierung

Die Software ist in einem detaillierten Modell beschrieben, das in einem Modellierungstool (Enterprise Architect) hinterlegt ist. Es beschreibt die Software von Kontext bis Paketebene. Pakete sind mit Zweck hinterlegt und enthalten wichtige Klassenkonstrukte.



www.embarc.de

Methodikkicker



Risikogetriebene Architektur

Die Architektur ist nicht in einheitlicher Detaillierung bearbeitet. Stattdessen bestimmen Risikoaspekte wie fehlende Erfahrung oder schwere Abbildbarkeit von Anforderungen den entgegengebrachten Aufwand und die Tiefe der Analyse und Dokumentation.



www.embarc.de

Methodikkicker



Standardisierung

Die Architekturbeschreibung folgt unternehmensweiten Standards. Gute Lösungsteile werden in eine Referenzbibliothek übernommen und stehen nachfolgenden Projekten zur Verfügung (potentiell samt Umsetzung).



www.embarc.de

Methodikkicker



Freiheit und Verantwortung

Teams dürfen lokal auch technologische Entscheidungen treffen – also die Datenbanktechnologie oder Plattform variieren. Abstimmungen mit anderen Teams gibt es vorrangig um Best-Practices zu kommunizieren.



www.embarc.de

Methodikkicker



Technische Schulden

In regelmäßigen Abständen wird nach technischen Schulden auf Code- und Architekturebene gesucht (Workshop). Die gefundenen Schwächen ergeben (Architektur-)aufgaben die je nach Priorität abgearbeitet werden.



www.embarc.de

Methodikkicker



Architektur-Board

Ein zentrales Gremium, bestehend aus erfahrenen Architekten, hat die Gesamtarchitektur im Blick. Weitreichende Entscheidungen, oder Änderungen/Ergänzungen am Technologiestack werden im Board beschlossen. Fehlentwicklungen werden beobachtet und korrigiert.



www.embarc.de

Methodikkicker



Quantitative Analyse

Mit der Hilfe von Analysetools wird der entwickelte Code analysiert und mit der geplanten Architektur verglichen (Structure101, Sonargraph). Strukturverstöße oder große bzw. komplexe Applikationsteile werden identifiziert und für Refactoring eingeplant.



www.embarc.de

Methodikkicker



Produktive Tests

Wichtige Qualitätseigenschaften der Applikation (Performanz, Lastverhalten, Sicherheit, ...) werden durch das Monitoring der Produktivumgebung überprüft. Fehlverhalten wird zeitnah korrigiert.



www.embarc.de

Projektkarte



Business System

zur Abbildung zentraler Geschäftsprozesse. Anwender sind qualifizierte Fachkräfte welche für die Nutzung geschult werden.

www.embarc.de

Organisationskicker



Die Jungen Wilden

Die jungen, hungrigen Entwickler innerhalb des Projekts experimentieren gerne und verfügen über breites, inhomogenes Know-How. Allesamt gute Leute...

1x 😞 2x 😞
5 ⌚ 3 ⌚

www.embarc.de

Organisationskicker



Management-Attention

Auf der Entwicklung des Produkts liegt die volle des Top-Managements. Der Erfolg des Projekts ist essenziell für das Unternehmen und jedes Experiment wird kritisch beäugt werden.

1x 😞 2x 😞
5 ⌚ 3 ⌚

www.embarc.de

Organisationskicker



Verteilte Entwicklung

Aus strategischen Erwägungen heraus findet die Entwicklung international verteilt an 4 Standorten statt.

1x 😞 2x 😞
5 ⌚ 3 ⌚

www.embarc.de

Organisationskicker



Weltweiter Einsatz und Support

Das Produkt soll international eingesetzt und unterstützt werden. Produkt-Features, Entwicklungs-, Betriebs- und Support-Prozesse sind darauf auszurichten.

1x 😞 2x 😞
5 ⌚ 3 ⌚

www.embarc.de

Organisationskicker



Zertifizierung des Produkts

Um das Produkt in regulierten Umfeldern vertreiben zu können, müssen der Entwicklungsprozess, die eingesetzte Werkzeuge sowie die erstellte Software selbst, vorgegeben Qualitätsrichtlinien entsprechen und regelmäßig auditiert werden.

1x 😞 2x 😞
5 ⌚ 3 ⌚

www.embarc.de

Organisationskicker



Time2Market

Um der Konkurrenz zuvor zu kommen soll das Produkt so schnell wie möglich zur Marktreife gebracht werden.

1x 😞 2x 😞
5 ⌚ 3 ⌚

www.embarc.de

Prozesskicker



Priorisiertes Product Backlog

Die Festlegung welche Features tatsächlich realisiert werden und an welchen als nächstes gearbeitet werden soll, erfolgt während des Projekts. Dazu wird ein Product Backlog gepflegt, in dem potentiell benötigten Anforderungen gesammelt und für eine schrittweise Realisierung heruntergebrachten werden.

3 ⌚ 1x 😞 2x 😞
2 ⌚ 0 ⌚

www.embarc.de

Prozesskicker



Informations-Radiator

Die Entwicklungsteams visualisieren an zentraler Stelle und für alle sichtbar wesentliche Projektinformationen, wie beispielsweise: Entwicklungsstand und aktuelle Probleme, Fehler-Level und -verlauf, anstehende Aufgaben, ...

3 ⌚ 1x 😞 2x 😞
2 ⌚ 0 ⌚

www.embarc.de

Prozesskicker



Terminplanung

Das Projekt hat regelmäßig einen Terminplan vorzulegen, in dem zu erkennen ist wie die weitere Planung aussieht, wann mit der Lieferung zu rechnen ist und ob es Verzug gibt.

3 ⌚ 1x 😞 2x 😞
2 ⌚ 0 ⌚

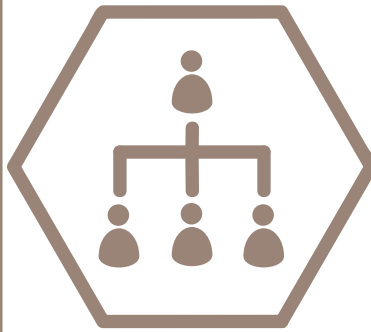
www.embarc.de

Organisationskicker



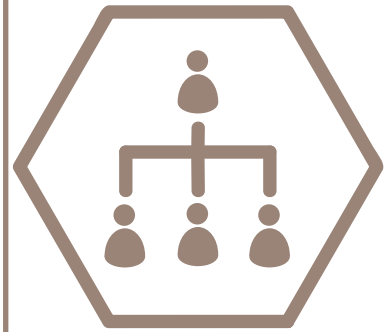
www.embarc.de 

Organisationskicker



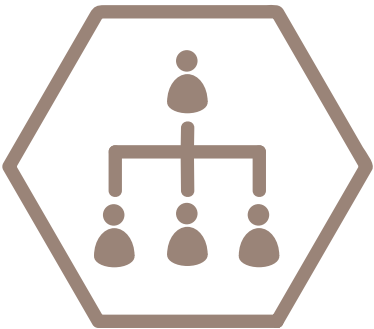
www.embarc.de 

Organisationskicker



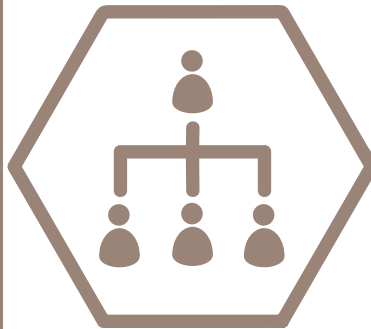
www.embarc.de 

Organisationskicker



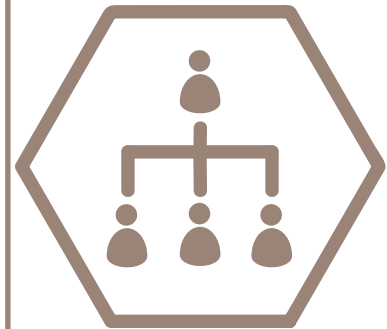
www.embarc.de 

Organisationskicker



www.embarc.de 

Organisationskicker



www.embarc.de 

Prozesskicker



www.embarc.de 

Prozesskicker



www.embarc.de 

Prozesskicker



www.embarc.de 

Prozesskicker



Management-Ampel

Dem Management wird im Rahmen des regelmäßigen Reportings mittels Ampelfarben signalisiert ob es Probleme gibt und Unterstützung benötigt wird.



www.embarc.de

Prozesskicker



Limitierung angefangener Arbeit (WIP-Limit)

Die Menge an Aufgaben die ein Team und/oder eine Mitarbeitender anfangen darf wird limitiert. Ist das Limit erreicht. Kann eine Aufgabe aufgrund von Problemen nicht weiter bearbeitet werden und das Limit ist erreicht, dürfen keine neuen Aufgaben angefangen werden.



www.embarc.de

Prozesskicker



Meilensteine

Mit den Stakeholdern werden Meilensteine vereinbart um gemeinsam strategische Entscheidungen über den weiteren Projektverlauf zu treffen und Projektrisiken neu zu bewerten.



www.embarc.de

Prozesskicker



Retrospektiven

Das Team reflektiert regelmäßig die Zusammenarbeit und seine Arbeitsprozesse und prüft wie es diese eigenständig verbessern kann.



www.embarc.de

Prozesskicker



Vierteljährliche Releases

Es gibt eine Vorgabe, dass Releases nur alle Vierteljahr in Produktion gehen um die Produktion nicht unnötig zu gefährden und Schulungsaufwände gering zu halten.



www.embarc.de

Prozesskicker



Stop&Fix

Sobald Fehler in einem integrierten System auftreten, ist sämtliche Entwicklungsarbeit einzustellen und alle konzentrieren sich auf die Identifizierung des Fehlers, seine Beseitigung, der Analyse wie diese Art Fehler entstehen konnte und was getan werden kann, damit diese Art Fehler in Zukunft nicht wieder entstehen kann.



www.embarc.de

Prozesskicker



Review

Auf Basis einer Live-Demonstration des aktuellen Entwicklungsstands des Systems reflektiert das Entwicklungsteam gemeinsam mit Stakeholdern potentielle Änderung am Scope sowie des weiteren Entwicklungsvorgehens.



www.embarc.de

Prozesskicker



Daily Stand-Up

Tägliches Treffen des Entwicklungsteams um sich über aktuelle Probleme und Entwicklungen auf dem Laufen zu halten und Schwierigkeiten rechtzeitig zu erkennen.



www.embarc.de

Prozesskicker



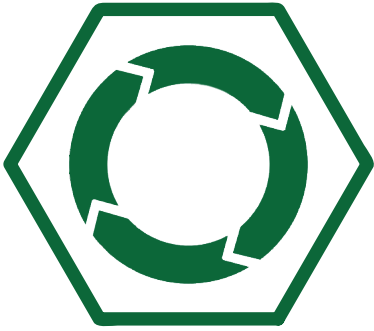
Autonome Tester

Test haben durch eine autonomes Testteam zu erfolgen um unabhängig und Objektivität sicherzustellen und nicht nur zu testen, was entwickelt wurde.



www.embarc.de

Prozesskicker



www.embarc.de 

Prozesskicker



www.embarc.de 

Prozesskicker



www.embarc.de 

Prozesskicker



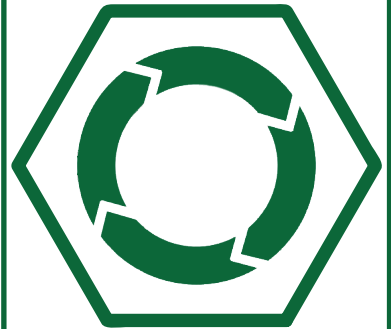
www.embarc.de 

Prozesskicker



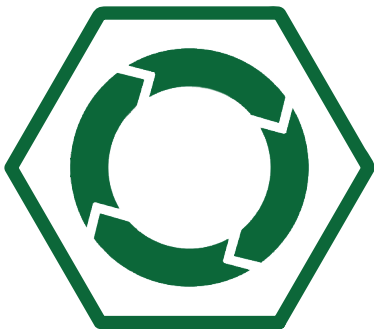
www.embarc.de 

Prozesskicker



www.embarc.de 

Prozesskicker



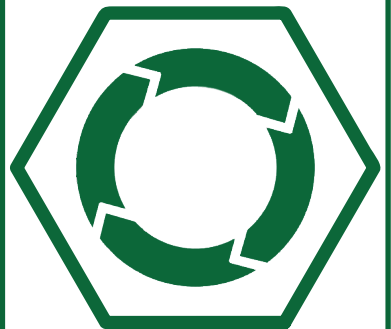
www.embarc.de 

Prozesskicker



www.embarc.de 

Prozesskicker



www.embarc.de 

Qualitätskicker



Audit-Trail

Es muss im System eindeutig nachvollziehbar sein welcher Benutzer welche Daten verändert hat und auch welche Daten er zu welchem Zeitpunkt gesehen hat.



www.embarc.de

Qualitätskicker



Korrektheit

Die fehlerfreie und korrekte Funktionalität der Anwendung ist unverhandelbar. Es darf keinerlei Fehler in der Produktionsumgebung/im Betrieb der Anwendung geben.



www.embarc.de

Qualitätskicker



Wiederverwendbarkeit

Lösungen fachlicher oder technischer Natur, die im Kontext der Anwendung erstellt werden, sollen mit möglichst wenig Aufwand in andere Anwendungen des Unternehmens integriert werden können.



www.embarc.de

Qualitätskicker



Modifizierbarkeit der Technologie

Neue Technologien, die auf dem Papier vielversprechend für die Evolution der Applikation aussehen, können sehr einfach getestet werden. Möglichst in der produktiven Umgebung.



www.embarc.de

Qualitätskicker



Modifizierbarkeit und Erweiterbarkeit

Werden Änderungsanforderungen oder Erweiterungswünsche gestellt, können diese sehr schnell bedient werden. Die entsprechenden neuen / angepassten Funktionen stehen möglichst bald zur Verfügung.



www.embarc.de

Qualitätskicker



Benutzbarkeit

Die gute Benutzbarkeit des Systems ist ein zentrales Unterscheidungsmerkmal von Anwendungen der Konkurrenz. Kurze Latenzzeiten und gute UX sind sehr wichtig.



www.embarc.de

Projektkarte



Krankenhaus-informationssystem

zur Erfassung, Verarbeitung und Weitergabe medizinischer und administrativer Daten im Krankenhaus. Umfasst Server-, genauso wie Arbeitsplatz-funktionen

www.embarc.de

Projektkarte



E-Commerce-System

zum Verkauf von Waren und/oder Dienstleistungen über das Internet

www.embarc.de

Projektkarte



Online-Gaming-Plattform

für browserbasierte und mobile Spiele nach dem Freemium-Prinzip, inklusive der dazugehörigen Community.

www.embarc.de

Qualitätskicker



www.embarc.de 

Qualitätskicker



www.embarc.de 

Qualitätskicker



www.embarc.de 

Qualitätskicker



www.embarc.de 

Qualitätskicker



www.embarc.de 

Qualitätskicker



www.embarc.de 

Projektkarte



www.embarc.de 

Projektkarte



www.embarc.de 

Projektkarte



www.embarc.de 

Qualitätskicker



Skalierbarkeit

Die Anwendung soll über gute Skalierungseigenschaften verfügen. Bei hoher Benutzer- oder Rechenlast wird das System nicht instabil oder unperformant. Bei niedriger Belastung werden möglichst wenige Ressourcen verschwendet.



www.embarc.de

Qualitätskicker



Ausfallssicheres System

Das System darf nicht ausfallen, muss 24/7 verfügbar sein. Im Zweifel sind Inkonsistenzen in Kauf zu nehmen um die Verfügbarkeit weiterhin zu garantieren.



www.embarc.de

Qualitätskicker



Offlinefähigkeit

Die Anwendung soll möglichst ohne Einschränkungen benutzt werden können, wenn die Internetverbindung gestört ist.



www.embarc.de

Qualitätskicker



Performanz

Das System zeichnet sich durch herausragendes Verbrauchs- und Zeitverhalten aus. Funktionalität an der Benutzer synchron interessiert sind, ist immer auf Effizienz optimiert. Datenzugriffe erfolgen so performant wie möglich.



www.embarc.de

Qualitätskicker



Wiederherstellbarkeit

Nach Fehlern welche die Benutzung des Systems beeinträchtigt haben, ist das System in der Lage einen konsistenten Zustand wiederherzustellen ohne dass eingegebene Daten verloren gehen.



www.embarc.de

Architekturstil



Microservices

Die Anwendung wird als Suite von relativ kleinen, technisch und fachlich unabhängigen Services entwickelt, die jeweils in einem eigenen Prozess laufen und über leichtgewichtige Mechanismen miteinander interagieren (etwa REST-calls).



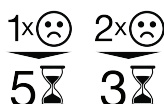
www.embarc.de

Architekturstil



Schichtenarchitektur

Die Anwendung wird in Schichten gegliedert, wobei eine definierte Schicht für die Abbildung der zentralen Geschäftslogik vorgesehen ist, und die anderen Schichten jeweils einen definierten technischen Zweck abbilden (Anzeige, Datenzugriff, Integration).



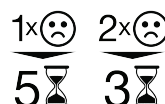
www.embarc.de

Architekturstil



Event-Getriebene Architektur

Die Anwendung wird als Set von fachlichen und technischen Komponenten entwickelt, die auf Events, wie Zustandsänderungen oder Aktivitäten anderer Komponenten, reagieren bzw. diese Events selbst produzieren. Zentrales Kommunikationselement ist ein Event-Bus.



www.embarc.de

Architekturstil



Big Ball of Mud

Die Anwendung wird ohne definierte Strukturierungsidee entwickelt. Generellen OO-Prinzipien werden beachtet, es gibt aber keine einschränkenden Zugriffsregeln für Pakete fachlicher oder technischer Natur. Kommunikation innerhalb der Anwendung ist vorwiegend synchron und direkt.



www.embarc.de

Qualitätskicker



www.embarc.de



Qualitätskicker



www.embarc.de



Qualitätskicker



www.embarc.de



Architekturstil



www.embarc.de



Qualitätskicker



www.embarc.de



Qualitätskicker



www.embarc.de



Architekturstil



www.embarc.de



Architekturstil



www.embarc.de



Architekturstil



www.embarc.de

