

Institut für Softwaretechnik (E188)
 Prüfung zur Vorlesung „Software Engineering“



TECHNISCHE
 UNIVERSITÄT
 WIEN
 VIENNA
 UNIVERSITY OF
 TECHNOLOGY

Matrikel-#	Familienname	Vorname	Kenn.-#

___ Zusatzblätter abgegeben.

___ Prüfungsantritt. Ich brauche das Ergebnis dringend bis ___.___ 2009 wegen _____

SEPM am 26.06.2009

	max. Pkt.	err.Pkt.
Wissens- und Verständnisfragen	60	
Arbeitsaufgaben	40	
<i>Summe</i>	<i>100</i>	

- Verwenden Sie nur Beiblätter der Größe A4. Schreiben Sie auf *ALLE Beiblätter* deutlich Ihren *Namen* und Ihre *Matrikelnummer*. Schreiben Sie die *Anzahl* der Beiblätter oben auf diese Seite.
- Sie haben für die Prüfung 120 min Zeit.
- Falls die Angabe bei einer Aufgabe Ihrer Meinung nach nicht eindeutig formuliert ist, treffen Sie selbst entsprechende als solche gekennzeichnete *schriftliche Annahmen*. Falls Sie Annahmen mit einem *Aufsichthabenden* abgesprochen haben, schreiben Sie dessen Namen zur Annahme.
- Bei der Prüfung sind **KEINE UNTERLAGEN** zugelassen!
- Das Prüfungsergebnis wird ca. 4 Wochen nach dem Prüfungstermin via TUWEL / TUWIS bekannt gegeben. Die Einsichtnahme findet nach persönlicher Terminvereinbarung statt.

Viel Erfolg bei der Prüfung.

WISSENS- UND VERSTÄNDNISFRAGEN

Projektmanagement (15 P)

1. Projektmanagement (7 Punkte).

- (a) Was ist ein Projekt und durch welche Merkmale ist ein Projekt gekennzeichnet? (3P)
- (b) Was versteht man unter einem Projekterfolg? Durch welche Rahmenbedingungen wird der Projekterfolg gefördert? Geben Sie für jede Rahmenbedingung zumindest ein konkretes Beispiel an. (4P)

2. Projektmanagement – Strukturpläne (8 Punkte).

- (a) Was verstehen Sie unter einem Projektstrukturplan (Work Breakdown Structure)? Aus welchen Ebenen besteht ein Projektstrukturplan? (2P)
- (b) Skizzieren und erläutern Sie einen (i) funktions- bzw. aufgabenorientierten Projektstrukturplan und (ii) einen objektorientierten Projektstrukturplan. Worin liegen die wesentlichen Unterschiede und wofür werden sie eingesetzt? (6P)

Life-Cycle Phasen und Prozessmodelle (15 P)

3. Software Prozesse: Agile Softwareentwicklung (8 Punkte)

- (a) Erläutern Sie anhand einer Skizze das Grundkonzept von SCRUM Sprints. (5P)
- (b) Erklären Sie den Unterschied zwischen dem Produkt-Backlog und Sprint-Backlog. (2P)
- (c) Was versteht man unter einem Burndown-Chart? (1P)

4. Software Wartung (7 Punkte)

- (a) Vervollständigen Sie die Tabelle um die Wartungskategorien, die in der Vorlesung besprochen wurden und geben Sie jeweils ein konkretes Beispiel an. (4P)
- (b) Welche Phasen umfasst der Wartungsprozess? (1P)
- (c) Nennen und beschreiben Sie zwei Techniken zur Wartung von Software. (2P)

	Correction	Enhancement
Proactive		
Reactive		

Technische Aspekte (15 P)

5. Architektur: DAO Pattern (5 Punkte)

- (a) Beschreiben Sie das Data Access Object Pattern (DAO) anhand eines UML-Diagramms und erläutern Sie dessen Funktionsweise. (3P)
- (b) Wann kommt dieses Pattern typischerweise zum Einsatz? (2P)

6. Traceability (5 Punkte).

- (a) Was verstehen Sie unter Traceability im Zusammenhang mit Softwareprojekten? (1P)
- (b) Beschreiben Sie kurz die unterschiedlichen Arten von Traceability und welche Vorteile sich in einem Softwareprojekt daraus ergeben. Wie kann Traceability in einem Softwareprojekt umgesetzt werden? (4P)

7. Integration (5 Punkte).

- (a) Erläutern Sie den Begriff der Systemintegration. (1P)
- (b) Skizzieren und erläutern Sie die vorgestellten Integrationsstrategien und welche Vor- bzw. Nachteile damit verbunden sind. (3P)
- (c) Was ist bei der Systemintegration im Hinblick auf die Qualitätssicherung zu beachten? (1P)

Qualitätssicherung (15 P)

8. Software Reviews (5P)

- (a) Was sind Software Reviews und wozu werden sie typischerweise eingesetzt? (1P)
- (b) Welche Arten von Reviews kennen Sie? Erläutern Sie die wesentlichen Aspekte der einzelnen Reviewarten und geben Sie jeweils ein konkretes Beispiel an. (2P)
- (c) Welche Rollen finden sich typischerweise bei Reviews und welche Aufgaben nehmen die unterschiedlichen Rollen wahr. (2P)

9. Test-Driven Development (5P)

- (a) Was versteht man unter Test-Driven Development (TDD)? (1P)
- (b) Beschreiben und skizzieren Sie die grundlegende Idee von TDD. (2P)
- (c) Welche Vorteile ergeben sich aus der Integration von TDD im Rahmen der Continuous Integration? (2P)

10. Qualitätssicherung und Testen (5 Punkte).

- (a) Diskutieren Sie die Begriffe Verifikation und Validierung. (1P)
- (b) Erläutern Sie die folgenden Test-Design-Techniken: Äquivalenzklassenzerlegung und Grenzwertanalyse. (2P)
- (c) Aus welchen Komponenten besteht eine Testfalldokumentation? Beschreiben Sie die notwendigen Komponenten und geben Sie ein konkretes Beispiel an. (2P)

Hinweis: Verwenden Sie eine tabellarische Darstellung zur besseren Strukturierung der Testfallbeschreibung.

ARBEITSAUFGABEN

Produktionsautomatisierung - Gemeinsame Information für alle Teilaufgaben:

Bei den Arbeitsaufgaben geht es um die Modellierung eines Softwaresystems aus verschiedenen Gesichtspunkten in einer automatisierten Fabrik. Die folgende Beschreibung (inklusive Abbildung) dient dem besseren Verständnis dieses Unternehmens. Relevante Informationen zu den Arbeitsaufgaben finden Sie direkt bei dem jeweiligen Unterpunkt jedes Beispiels.

Traditionelle Fabriken (Fertigungseinrichtungen) stellen große Mengen gleicher Produkte her, um über die Menge die Produktionskosten gering zu halten. Wesentlicher Teil der Kosten sind Aufwände für die Umstellung (Umrüstung) der Maschinen für neue Produkte oder Produktversionen. In der vorliegenden Fabrik sollen auch Kleinserien kostengünstig und flexibel herstellbar sein. Daher ist der Einsatz von mehreren flexiblen Maschinen erforderlich. Jede Maschine kann grundsätzlich mehrere Funktionen erfüllen, ist jedoch für eine definierte Funktion konfiguriert. Eine Änderung der Funktion der Maschine erfordert eine Umrüstung (Änderung der Softwarekonfiguration und gegebenenfalls von Hardwarekomponenten). Eine Funktion umfasst dabei die Herstellung bzw. den Zusammenbau komplexerer Produkte aus mehreren einfacheren Basisprodukten (Assembling).

In der Fabrik existieren zwei zusammen arbeitende Systeme, um Aufträge zu planen und die verteilten Maschinen zu steuern: Ein *Dispatcher (Lastverteiler)* stellt aus hereinkommenden Aufträgen konkrete Arbeitsaufträge für die Schicht zusammenstellt und ein *Produktionssystem*, das die verteilten Knoten des Systems (Maschinen, Speicher, Transportsystem) koordiniert, um Aufträge effizient abzuarbeiten. Ausfälle von Teilsystemen werden soweit möglich automatisiert korrigiert (z.B. durch redundante Transportwege oder redundante Maschinen).

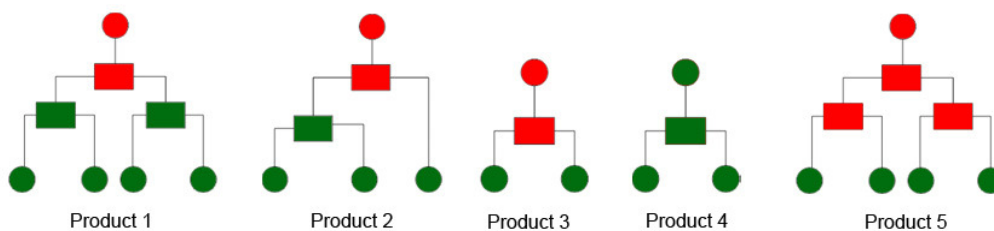
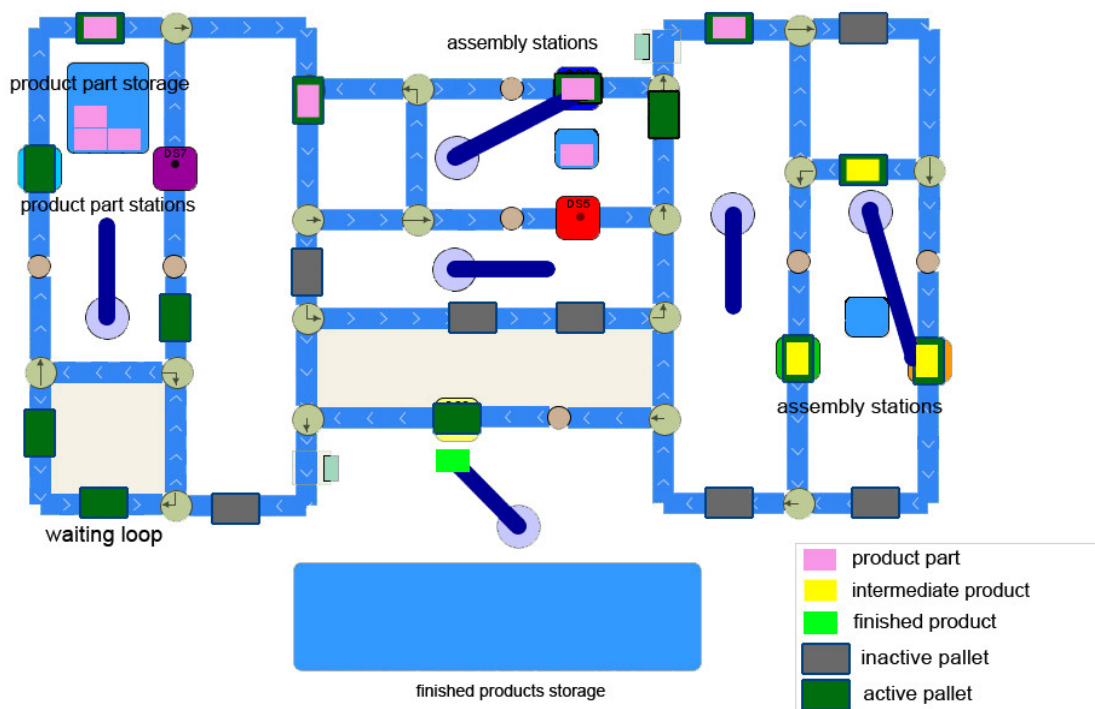


Abbildung: Workshop-Layout eines Produktionssystems mit Produktplänen.

Die Abbildung zeigt schematisch ein Aufbau der automatisierten Fabrik (*Workshop Layout*) mit folgenden Komponenten:

- *Product Part Storage*: Links oben befindet sich das Lager mit einfacheren angelieferten Basisprodukten und Zwischenprodukten aus der aktuellen Produktion.
- *Assembly Station*: In der Mitte oben und rechts befinden sich Maschinen (Rechtecke mit runden Ecken), die bestimmte Funktionen (etwa Schrauben, Kleben, Nieten) erfüllen können.
- *Finished Product Storage*: In der Mitte unten befindet sich das Lager für fertige Produkte, die auszuliefern sind.
- *Transportsystem*: Die Lager und Maschinen sind durch ein Transportsystem verbunden, das aus Förderbändern, Kreuzungen und Paletten besteht. Paletten (größere grüne oder graue Rechtecke) sind leer oder können Produkte (kleinere rosa, gelbe, grüne Rechtecke) befördern. Förderbänder (blaue Bänder mit Richtungspfeilen) können Paletten in eine Richtung bewegen. Kreuzungen (graue Kreise mit Richtungspfeil) können Paletten bei mehreren Ausgängen in einen Ausgang leiten oder die Palette aufhalten. Vor jeder Kreuzung gibt es einen Sensor, der das Herannahen einer Palette anzeigt und auch die Ladung der Palette (via RFID Chip) liest.

Der untere Teil der Abbildung zeigt mehrere einfache Produktpläne, die generell Baumform haben: Oben an der Spitze steht das Endprodukt (Kreis), unten an der Basis die Eingangsprodukte (Kreise). Um ein Endprodukt herzustellen, sind die Eingangsprodukte durch Maschinenfunktionen (Rechtecke) zu kombinieren (etwa Zusammennieten von mehreren Teilen). In der Abbildung werden in jeder Funktion zwei einfachere Teile zu einem komplexeren Teil zusammengebaut. Falls mehrere Maschinenfunktionen notwendig sind, um das Endprodukt herzustellen, sind die Zwischenprodukte geeignet weiter zu transportieren. Dabei ist es wesentlich, dass zu einer Maschine, die eine passende Funktion ausüben kann, die passenden Teile in der richtigen Reihenfolge ankommen. Die Produktpläne zeigen den Arbeitsfortschritt von aktuell bearbeiteten Produkten.

11. Domänenmodell (15 Punkte)

Ein bestimmtes Workshop Layout (Aufbau der automatisierten Fabrik) besteht aus einer Reihe von verschiedenen Komponenten. Jede dieser Komponenten wird durch eine eindeutige ID identifiziert, und außerdem durch x- und y-Koordinaten im Gesamtlayout positioniert.

Der Einfachheit halber treten in diesem Workshop nur folgende Komponenten auf:

- Fließbänder mit einer bestimmten Geschwindigkeit.
- Kreuzungen, die mehrere Fließbänder verbinden oder teilen.
- Paletten, die entweder leer oder mit einem Produkt beladen von den Fließbändern befördert werden.
- Maschinen, die eine bestimmte Reihe von verschiedenen Funktionen anbieten (z.B. Schweißen, Kleben) und eine bestimmte momentane Auslastungsrate besitzen.
- Lager mit einer bestimmten Kapazität, in denen Produkte gelagert werden können.

Für eine bestimmte Produktionsschicht wird ein festgelegtes Workshop Layout benutzt, außerdem besitzt eine Schicht einen vorher festgelegten Start- und Endzeitpunkt. In einer Schicht wird eine bestimmte Anzahl von Arbeitsaufträgen ausgeführt. Jeder Arbeitsauftrag hat ein Fälligkeitsdatum und wird aus einer Bestellung eines Kunden abgeleitet. Das bedeutet, dass für Bestellung mit einer eindeutigen Bestellnummer, einer eindeutigen Kundennummer, einer Liste an bestellten Produkten und einem Fälligkeitsdatum, ein oder mehrere Arbeitsaufträge erstellt werden, welche dann im Rahmen einer Produktionsschicht abgearbeitet werden.

Ein Produkt wird zusammengesetzt aus einem oder mehreren Subprodukten, welche wiederum ein Produkt sind. Zur Zusammensetzung ist außerdem eine bestimmte Anzahl an Maschinenfunktionen nötig.

Entwerfen Sie ein Domänenmodell in UML, das die erforderlichen Daten beinhaltet und die beschriebenen Abläufe unterstützt. Achten Sie darauf, die Systemkomplexität durch die Verwendung möglichst weniger Klassen und durch die Vermeidung unnötiger Redundanzen gering zu halten. Führen Sie bei jeder Klasse die zugehörigen Attribute mit Typen an. Geben Sie jeder Assoziation einen lesbaren und eindeutigen Namen sowie Multiplizitäten; überprüfen Sie, zu welchen Beziehungen Attribute zu speichern sind (und geben Sie diese als Assoziative Klassen an).

12. IDEF0 / Aktivitätsdiagramm (15 Punkte)

Der Dispatcher wählt aus hereinkommenden Aufträgen (Endprodukten) und verfügbaren Basis- und Zwischenprodukte im Eingangsspeicher besonders wertvolle Aufträge aus welche jedenfalls in der aktuellen Schicht durchzuführen sind. Aus den Produktplänen und den bereits erfolgten Aufträgen kann der Dispatcher anschließend berechnen, welche Maschinenfunktionen wie oft benötigt werden, um eine Menge von Aufträgen zu erfüllen.

Weiter weiß der Dispatcher, welche Funktionen die Maschinen im Produktionssystem zur Verfügung stellen, wie lange eine Maschine braucht, um eine Funktion zu erledigen und wie lange im Durchschnitt der Transport zwischen zwei Punkten im Produktionssystem dauert. Aus diesen Daten kann der Dispatcher für eine Schicht mit bekannter Dauer berechnen, welche Aufträge in dieser Schicht mit hoher Wahrscheinlichkeit ausgeführt werden können und wie hoch die Maschinenauslastung ist.

Die Zusammenarbeit zwischen Dispatcher und Produktionssystem erfolgt über den Austausch von Nachrichten: Sobald zumindest eine Palette verfügbar ist, übergibt das Produktionssystem diese Information dem Dispatcher, welcher die aktuell wichtigste Aufgabe (z.B. Holen von Produkten, Ausführen einer Maschinenfunktion, Speichern von Produkten) an das Produktionssystem liefert. Das Produktionssystem meldet dem Dispatcher die Kapazität an verfügbaren Paletten und erfolgte

Aufträge sowie Fehler, welche die Kapazität des Systems bzw. den Erfolg von Aufträgen betreffen. Der Dispatcher bewertet die möglichen Aufträge und schickt Sie an das Produktionssystem, um für die verbleibende Restdauer der Schicht möglichst wertvolle Aufträge erfolgreich zu beenden.

Aktivitäten des Dispatchers:

- Auswahl besonders wertvoller Aufträge
- Berechnung der benötigten Maschinenfunktionen.
- Priorisieren von Aufträgen für Maschinen mit hoher Auslastung.
- Bewertung der möglichen Aufträge

Bilden Sie die oben beschriebenen Abläufe für den Dispatcher als Aktivitätsdiagramm in IDEF0 Notation ab.

- **Kontrollfluss (5 Punkte):**
Bilden Sie den Kontrollfluss des Dispatchers ab.
- **Datenfluss (10 Punkte)**
Beschreiben Sie die wesentlichen Ein- und Ausgabeparameter sowie Kontrollparameter und etwaige Ressourcen welche notwendig sind um den Kontrollfluss zu gewährleisten.

13. Zustandsdiagramm (10 Punkte)

Jede Maschine kennt die Teile, die für eine bestimmte Funktion gebraucht werden. Sobald ein passender erster Teil zur Maschine kommt, stoppt die Palette bei der Maschine, die Maschine nimmt das Produkt von der Palette, und die Palette fährt weiter. Danach wartet die Maschine auf den passenden nächsten Teil, um ihre Funktion auszuführen. Paletten, die keine passenden Teile haben, werden nicht angehalten, sondern fahren weiter. Die Palette mit dem letzten passenden Teil für eine Funktion wird solange angehalten, bis die Maschine ihre Funktion erfüllt hat und das Ergebnisprodukt auf die wartende Palette legt.

Bilden Sie die Zustände einer Maschine des Produktionssystems ab, verwenden Sie dazu UML Notation. Jede Maschine ist vorkonfiguriert ein bestimmtes Produkt und einen bestimmten Arbeitsschritt zu vollziehen.