Example solutions to exam 2015-10-30 in reverse order

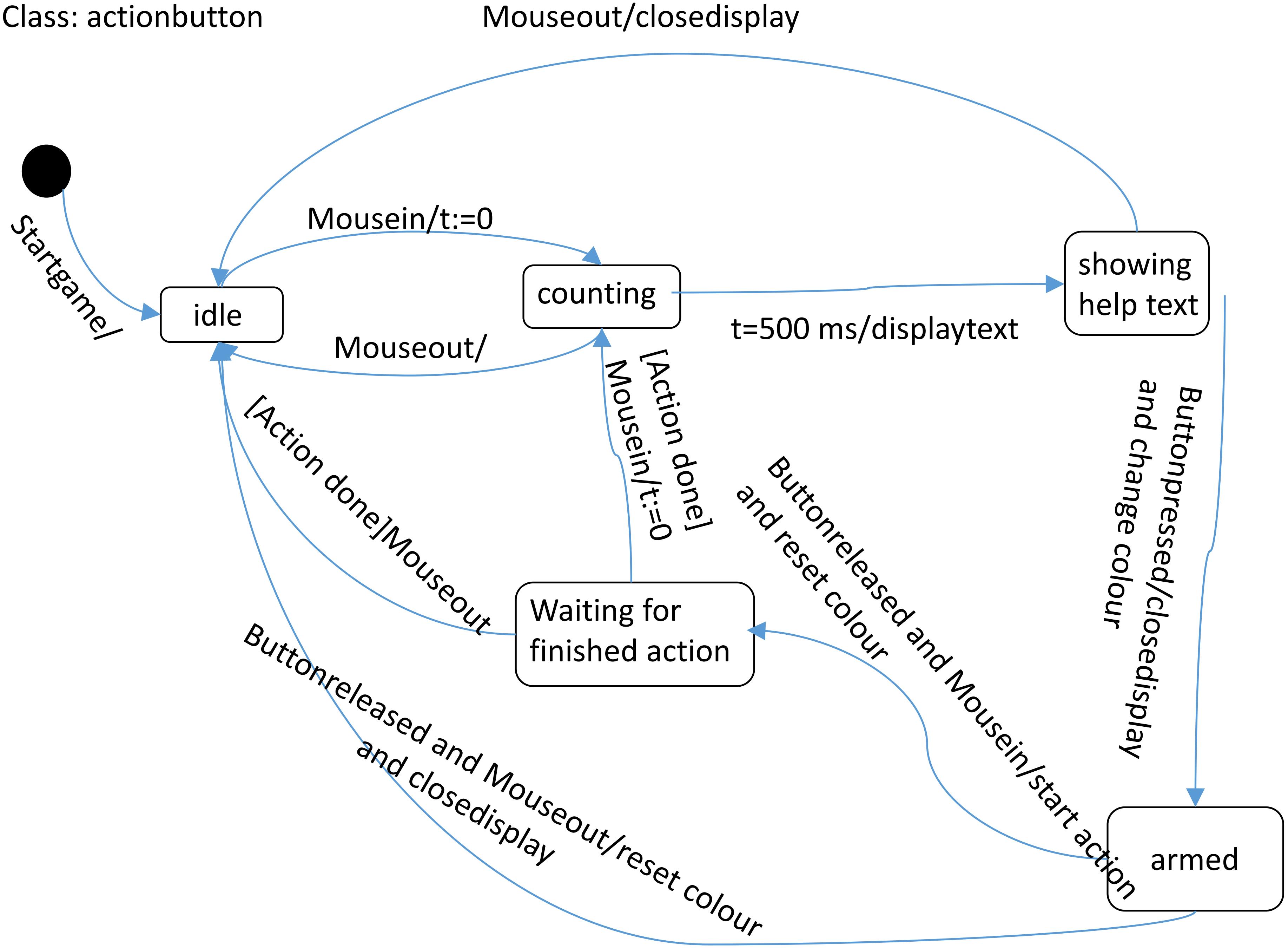
As I said during the lecture, there are many different solutions possible. Misspellings and grammar errors do appear

8a

Example of elaborate solution:

Assumption: the button changes colour when armed
We assume that the time resolution of the clock is good enough
We don’t care about what happens when ending the game
t = the value of a timer that can be started and reset
Displaytext = the correct text of the button is shown
Closedisplay = the text is no longer shown
Mousein = the position of the mouse is within the area of the button
Mouseout = the position of the mouse is outside the area of the button

Class: actionbutton



Mouseout/closedisplay

Startgame/

idle

Mousein/t:=0

counting

t=500 ms/displaytext

showing help text

Mouseout/

[Action done]

Mousein/t:=0

[Action done]Mouseout

Waiting for finished action

Buttonreleased and Mousein/start action

and reset colour

Buttonpressed/closedisplay

and change colour

armed

Buttonreleased and Mouseout/reset colour

and closedisplay

8a

Grading

Start with 10 cr

-1 for UML syntax problem

-1 for UML semantic problem

-1 for wrong or missing behaviour from the text

-1 per necessary assumption stated wrongly or left out

-2 if states of several classes

We accept ovals as state symbols.

We do not accept sharp corner rectangle.

It’s OK to leave “/” out if there are no actions

8b

Input parameters: mouseposition (MP): {in, out}
Button(B): {pressed, released}
Time since last MP change (T): integer
Intermediary variable: Action done (AD): Boolean
Output parameters: displaytext(D): boolean
colour (C): {default, changed}
Action started (AS): boolean

state = current state
state’ = next state

Assumptions: game has started, we can change and set stop-point of timer, states can be observed

8b

Test suite 1

no	MP	B	T(ms)	AD	D	C	AS	state	state’
1	in	released	0	-	false	default	false	idle	counting
2	in	released	500	-	true	default	false	counting	showing…
3	in	pressed	-	-	false	changed	false	showing…	armed
4	in	released	-	-	false	default	true	armed	waiting…
5	in	released	1500	true	false	default	false	waiting	counting
6	out	released	200		false	default	false	counting	idle

8b

Test suite 2

no	MP	B	T(ms)	AD	D	C	AS	state	state’
1	in	released	0	-	false	default	false	idle	counting
2	in	released	500	-	true	default	false	counting	showing…
3	in	pressed	-	-	false	changed	false	showing…	armed
4	in	released	-	-	false	default	true	armed	waiting…
5	out	-	-	true	false	default	false	waiting…	idle

8b

Test suite 3

no	MP	B	T(ms)	AD	D	C	AS	state	state’
1	in	released	0	-	false	default	false	idle	counting
2	in	released	500	-	true	default	false	counting	showing…
3	in	pressed	-	-	false	changed	false	showing…	armed
4	out	released	-	true	false	default	false	armed	idle

8b

Test suite 4

no	MP	B	T(ms)	AD	D	C	AS	state	state’
1	in	released	0	-	false	default	false	idle	counting
2	in	released	500	-	true	default	false	counting	showing…
3	out	released	-	-	false	changed	false	showing…	idle

8b

Grading

Start with 10 cr

-1 per missing parameter

-1 per missing transition

-1 for missing expected output

-2 for no explicit reference to transition

-1 for necessary assumption left out or wrongly stated

Principle I:

a) 1. Involve the customers to evaluate and prioritize work frequently. So that things that are in the focus of the customer are prioritized.

2. Organize work in fairy short sprints, so you don’t invest too much time in implementation without checking with the customer.

3. Automate as much of the testing and integration as possible. This is to save time to allow for short sprints, and save the customer from finding more or less trivial bugs.

4. Apply continuous integration and/or deployment in each iteration

b) In the waterfall model the customer gets a large delivery at the end of the project. If the project is long there is a risk that wrong design decisions are not found after a large investment in coding is done.

c) There is always the risk that the customer is not competent in giving good feedback.

7

Principle IV:

a) 1. Business people can be present at daily meetings – then they will be updated and known in the development team.
   2. Give business people access to the latest running version - then they can be informed and trained to do demos.
   3. Involve business people in testing, also if there is a customer available. They will provide a more general view of the product.

b) In the Waterfall model, this is only done in requirements and acceptance testing phases.

c) Business people are often extrovert and good at talking and their knowledge about design can be overestimated.

7

Principle VII:

a) 1. Use estimation and burn-down chart to find the velocity of the team that works for 40-hour weeks.

2. Integrate and test often so that you don’t build up a backlog of testing work and bugs.

3. After the daily meeting, make sure that reported problems get attention early by the whole team, so that no single person has to struggle without using the competence in the team.

4. Developers estimate time (using for example planning poker)

5. Sprint retrospective at the end of the sprint to be better for the following sprints

b) In the Waterfall model, there will be busy periods, and since everyone knows this it will become a self-fulfilled prophesy.

c) It is a risk that a team that is never challenged will not develop their competence.

7

Principle X:

a) 1. Use story cards to make clear exactly what the customer would like to see. If requirements are vague, there is a risk of unnecessary work.

2. Only implement what is in the requirements and demo that, if the customer would like to modify you have not spoiled a large investment.

3. Use test-driven development to make sure that you develop towards a fixed target instead of spending time doing what you think might be needed to pass unit testing.

5. Sprint retrospective at the end of the sprint to be better for the following sprints

b) In the waterfall model you invest much into a maintainable system, which is good, but there is a risk of waste if you need to change requirements. More focus on documentation as replacement for communication.

c) The risk is that you end up with a system that has performance problems since you have taken small steps only with functional requirements.

7

Distribute 5 credits per principle: 3 for the a) questions, and one each for b) and c)

Students can refer to different parts of the text, for instance, that a risk is the same for two principles.

6

Risk1: the teams work in different directions
Probability 4 (it is hard to think and prioritize in the same way)
Impact 2 (with a good interface, things can be repaired)
Plan: Mitigation: Integrate and test both teams often.
Make study visits at each other's sites.

Risk2: the end-user is stuck due temporary hinders
Probability 3 (larger cities require much infrastructure maintenance)
Impact 5 (this can be a real show-stopper if repeated and rumor is spread)
Plan: Transfer: Cooperate with a local vendor with a god history of updating local accessibility database.

Disclaimer: This page contains imaginary risks with the purpose of illustrating the process.

Risk3: The database expert might quit if she builds a family
Probability: 4 (typical for people of her age, the tradition is strong)
Impact: 4 (a key worker is hard to replace, you will lose lead-time)
Plan: Mitigation: Develop a support program for women to return after childbirth together with local expertise

Risk 4: The Phone Gap solution is beaten by competitors' native apps.
Probability: 3 (there are always niche developers waiting for a breakthrough who put lot of overtime)
Impact: 2 (the customer base might shrink amongst demanding users, multi-platform is still a plus)
Plan: Contingency plan: Have experimental prototypes ready, eg. by master's theses of the UX designer's students

Disclaimer: This page contains imaginary risks with the purpose of illustrating the process.

6 Risk5: NFC will be delayed for iPhone
Probability: 5 (historically Apple is unpredictable)
Impact: 5 (disappointed customers)
Plan: Avoidance: Don’t release for iPhone until legally binding information is obtained.

2 credits per sensible risk, maximally 10 credits. 1 credit for a description with short motivation for Probability and Impact. 1 credit for a motivated Plan.
We might deduct some credits from the totally earned if there are systematic weaknesses.
Students are allowed to write about general risks, as long as they are well motivated.

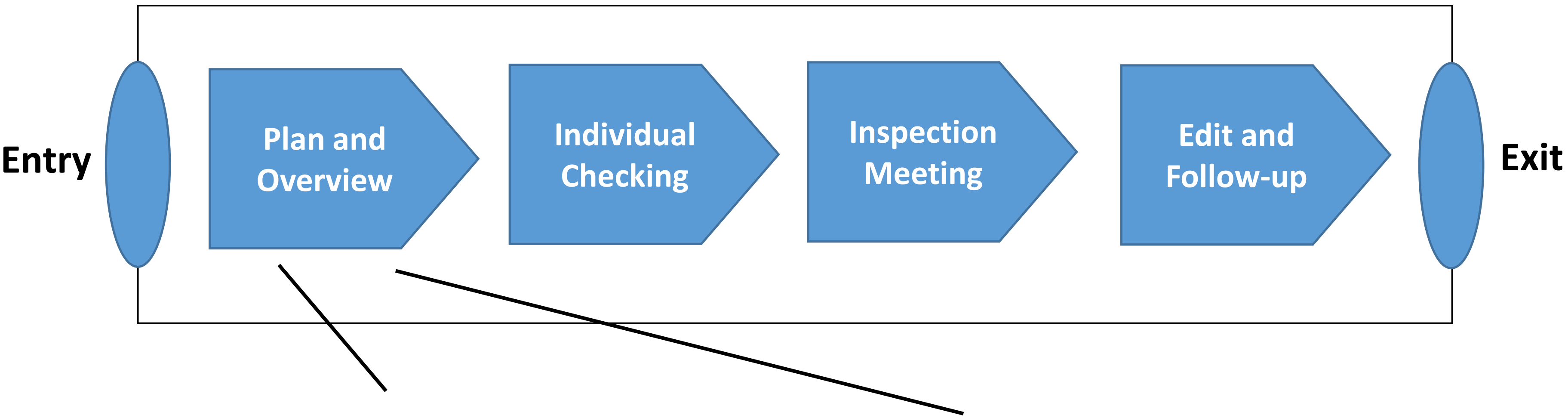
Disclaimer: This page contains imaginary risks with the purpose of illustrating the process.

5b

Project Planning:
The purpose is to write and maintain a plan to define the project. Important factors to consider when developing the plan: estimation, scope, stakeholders resources, risks. There shall be a commitment in the entire organization to the project plan. Whenever needed the project plan is revised. The plan doesn’t need to be a single document, it can be distributed to different documents and tools.

This will be good to my company since time is less likely to slip with a well-maintained plan that everyone is following. When planning it is a good opportunity to think of what can be an expected. This is extra valuable if the customer participates in the planning.

Inspection Process



Inspection leader

Planning the inspection

- Identify inspection team
- Assign responsibilities
- Schedule meetings
- Distribute material
- Specify scope and priorities

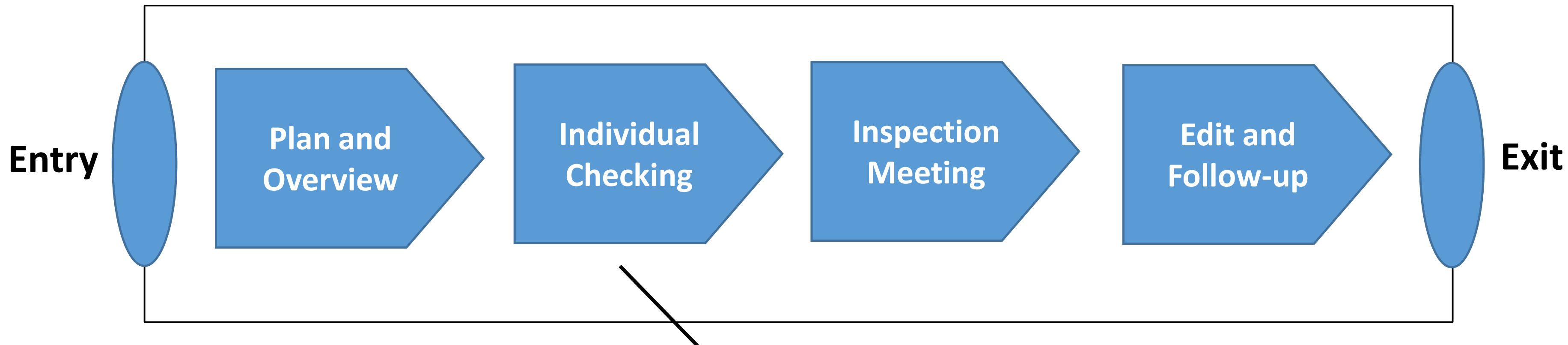
Overview

- Introduce the product

Author

5c

Inspection Process



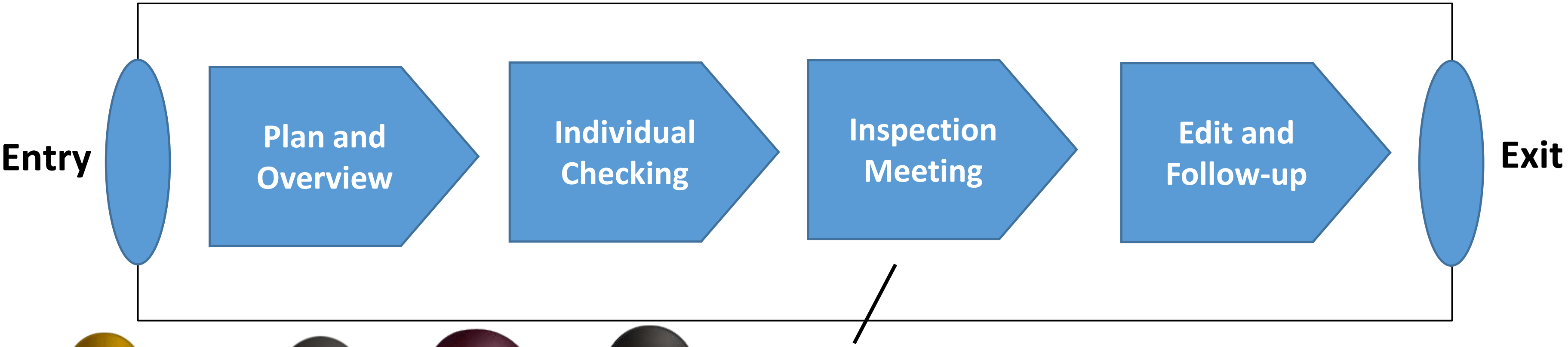
Individual checking

- Exam the product individually
- Report all defects to the inspection leader
- Prepare for the inspection meeting

Inspection rate (IEEE Std 1028-2008)

- Requirements or Architecture (2-3 pages per hour)
- Source code (100-200 lines per hour)

Inspection Process



Meeting agenda

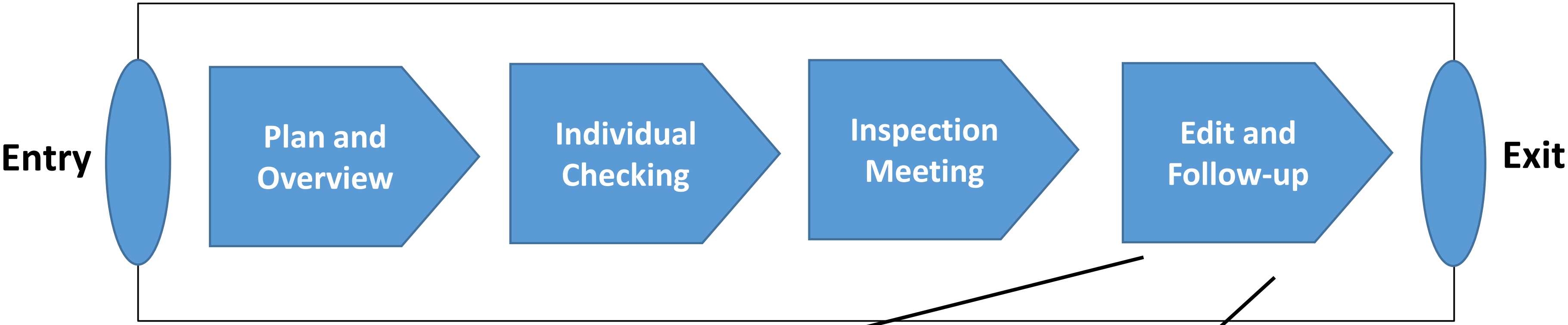
- Introduction of roles and purpose
- Reader presents the product (details)
- Inspect product, produce defect list (whole team)
- Review defect/anomaly list (completeness and accuracy)
- Make exit decission

Should detect, not resolve defects

Exit decisions (1, 2 or 3)

1. Accept with no further verification
2. Accept with rework verification (verify by one member)
3. Reinspect – redo the the process

Inspection Process



Entry

Plan and Overview

Individual Checking

Inspection Meeting

Edit and Follow-up

Exit

Edit

- Author resolves items

Author

Follow-up

- Inspection leader verifies that all items are closed

Inspection leader

4b

Error: A human mistake that makes the human programmer to write the wrong program.

Fault: A part of the program that is not correctly implemented.

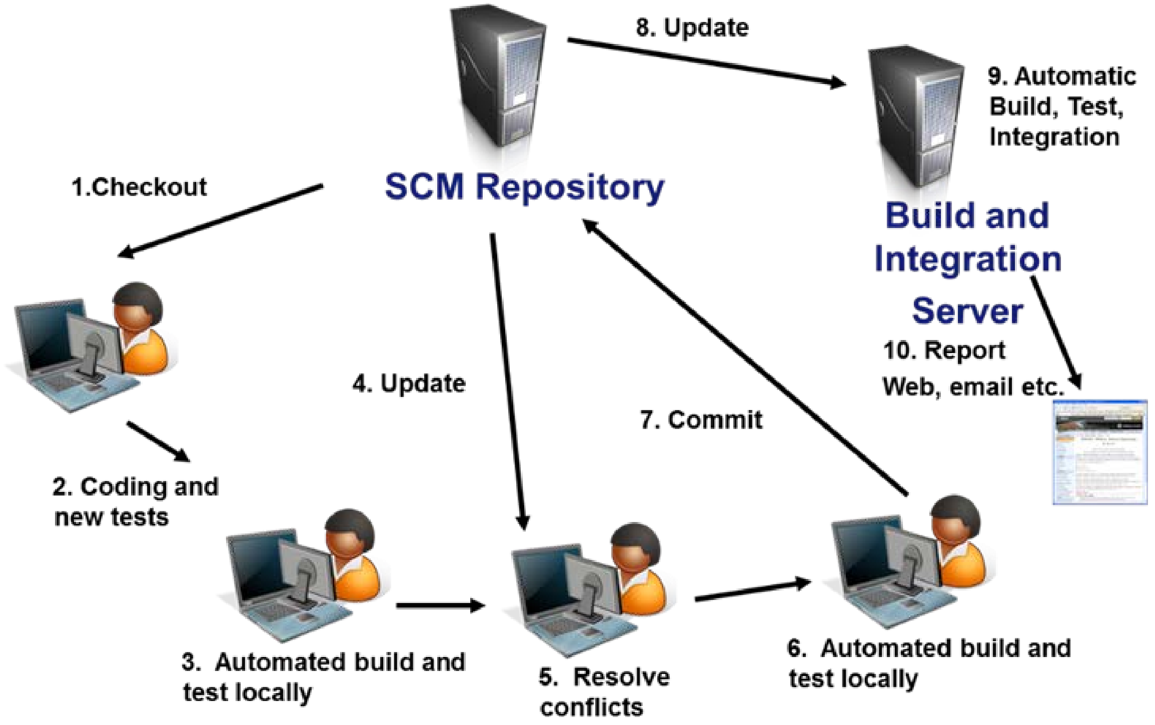
Failure: An event where the program is executed in such a way that unintended behavior is visible.

Oracle: A human or a system that determines if a test case did pass or not.

4c

16

Continuous Integration - Workflow



SCM Repository

Build and Integration Server

1.Checkout

2. Coding and new tests

3. Automated build and test locally

4. Update

5. Resolve conflicts

6. Automated build and test locally

7. Commit

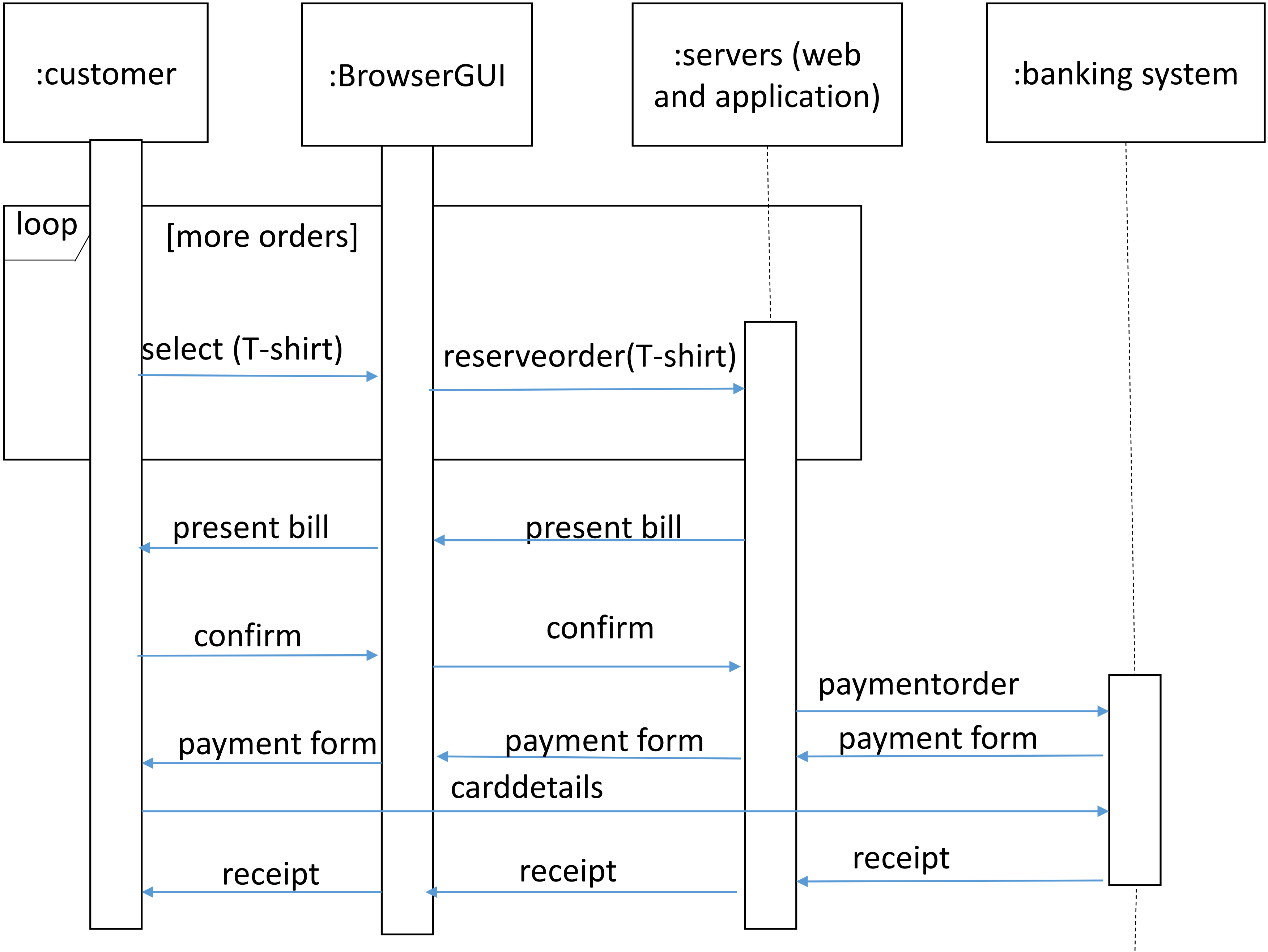
8. Update

9. Automatic Build, Test, Integration

10. Report Web, email etc.

4 cr for a good solution
-2 if the build and integration server is forgotten
-1 per forgotten activity

3b


:customer
:BrowserGUI
:servers (web and application)
:banking system
loop
[more orders]
select (T-shirt)
reserveorder(T-shirt)
present bill
present bill
confirm
confirm
paymentorder
payment form
payment form
payment form
carddetails
receipt
receipt
receipt

3c

Idea 1: Isolate the safety-critical parts into a single subsystem, and use advanced methods and experienced personnel in the design. Thereby we can justify the usage of expensive techniques for the subsystem.

Idea2: Specify several variants for the same function. Use some kind of voting mechanism to determine if one variant is giving deviant results. With this approach we can protect ourselves from letting programming errors cause a failure.

2 cr for a viable idea
2 cr for a motivation

2b

Toll-gate – a decision point where the customer or a product committee decides about the future of the project. Normally this includes selecting features to continue with.

Mile-stone – a decision point where the status of the project is determined. Normally this means to evaluate the project progress against specific goals.

2 cr per sensible explanation

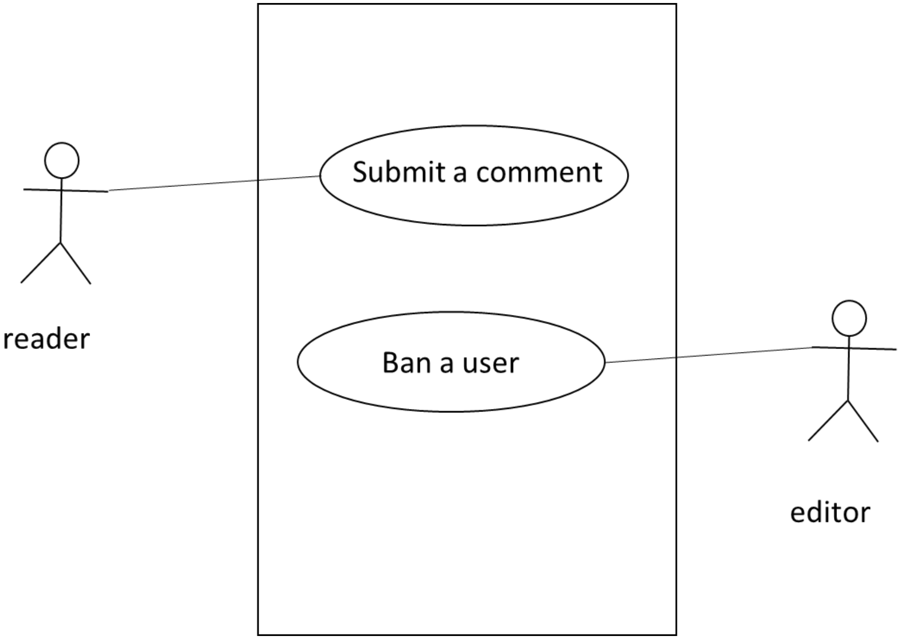
2c

Stakeholder – A person or organization that is affected by the outcome of the project. They might be direct, such as an end-user, or indirect, such as the customer's customer.

Example of a student portal:
The students – want to access the information and services
Education administrators – want to provide information and services
Teachers – are dependent of correct information sent to students
The university leaders – need to have a rational way of administrating studies
LiU IT – creates and maintains the application software and infrastructure

1 cr for a correct definition
1 cr for a stakeholder, max 3 cr
Except for the students, we need some little motivation for why they are stakeholders

1b



Submit a comment:

The reader of the newspaper logs into the system with his/her credentials.
The reader browses the web-version of the newspaper and clicks the “comment” button.
The reader is directed to an input-page where he/she formulates the comment.
The reader clicks the “submit” button and is directed to a confirmation page.
The reader clicks OK and is directed back to the article.

Ban a user:

The editor receives a complaint message over mail.
The editor logs in with his/her credentials.
The editor browses the comments to find out the object of the complaint.
The editor removes the comment.
The editor checks the last-warning file for the readers.
The editor enters the user registry and searches the failing user.
The editor marks the user and clicks the “remove” button.
The editor is directed to the remove page and writes a text to be sent to the failing user.
The editor clicks the “confirm” button and is directed back to the article.

1b

Two actors and two ovals with a verb phrase plus a system boundary are needed in the diagram.
Texts of 3-5 sentences similar to the examples are needed.
2p per good use-case
at least 3 sentences per use-case
actors are roles, not user1 and user2
actor can be a sub-system
Two single use-case diagrams minus 2 credits.
Only a correct diagram 1 credit. Diagram missing minus 1 cr.
Use-case name: a verb phrase

1c

Design constraint – A requirement that limits the way the system can be implemented. For instance, “The system shall be implemented in Java.”

Quality requirements – Requirements about measurable goals for different quality factors. For instance, “The system shall respond to user input in less than 0,5 seconds”

Requirements elicitation – The process where an analyst gets an understanding of the true needs of the customer. The result of an elicitation process need not to be written down. For example, the analyst interviews the customer about needs.

Requirements validation – The process of making sure that the requirements represent the true needs of the customer. For example, inspection with experts, simulation etc.

1 cr per sensible explanation. Examples are not needed

Questions and Answers Session

During your study many questions might arise. Collect your questions and come to this occasion.

Wednesday, October 26, 13:00-15:00

Alan Turing E-house, floor 3

Pass condition

Part I: Fundamentals

- Requirements
- Planning and Processes
- Design and Architecture
- Testing and SCM
- Software Quality

10 credits per area. Max 50 credits.

Part II: Advanced

50 credits, distributed over 2-5 questions.

- argue, compare, and analyze different concepts and techniques.
- construct and/or design solutions to larger problem.
- explain more advanced and specific topics.

To pass the exam (alternatives)

1. a) at least 4 credits in all areas in fundamentals **and**
   b) at least 50 credits in total
2. a) at least 4 credits in at least 4 areas **and**
   b) at least 60 credits in total

Grades if pass condition is met

Total credits	Mark
0-49	U
50-66	3
67-83	4
84-	5

Congratulations on 25 years of service

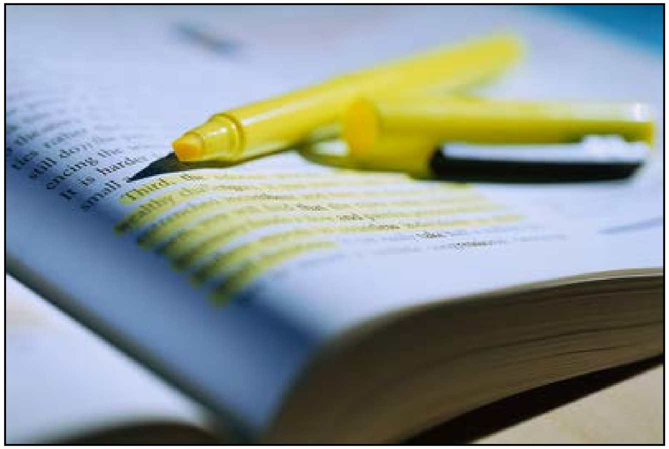
This certifies that

has dedicated twenty five years of service.

Aids

Allowed aids

- Two sheets of **handwritten** A4 papers (can write on both sides)
- One volume of dictionary to or from English or an English wordbook.



Explicitly forbidden aids

- Textbook
- Machine-written pages
- Photocopied pages
- Pages of other format than A4
- Electronic equipment

Hints

- Register for the exam
- Never guess on two alternatives of multiple-choice questions
- Use a pencil
- Use ergonomic aids
- Have the nerve to read through the exam first
- Use time-boxing and buffer time
- Do as the exam vigilators say

Thanks for listening!

me, when all students pass