ID2207- Modern Methods in Software Engineering

Homework 2

Course Coordinator:

Mihhail Matskin

Course Assistant:

Shatha Jaradat

Óttar Guðmundsson and Örn Arnar Karlsson

2017.9.17

1. Glossary for the problem

- **Client Request Detail:** A form filled during the first business meeting, where the client requests and preferences are filled in.
- **EventHomePage:** For each event EventSystem creates a home page. Every party that is involved in the event has access to the the EventHomePage but it does look different for people that have different responsibilities.
- **EventPlan:** List of tasks needed to do for the event to take place.
- **EventReport:** Report that is generated by AdministrationTeam after each Event is held.
- **EventRequest:** An object in EventSystem that is created by CustomerService.
- **EventNotes:** Bound to each approved EventRequest, these are notes that are written down on the FirstBusinessMeeting.
- **EventSystem:** The system the is going to be developed will be referenced hereafter as such.
- **EventStaffing:** File bound to EventRequest where notes regarding what staff is working and how long.
- **FirstBusinessMeeting:** The first meeting that is held after SeniorCustomerService accepts EventRequest.
- **Fileserver:** External server hosted at remote service where information regarding clients, events and EventReports are kept.
- **OtherBusinessMeeting**: Business meeting that might occur later when negotiating budget or make deals regarding the theme.
- **Production Sub-Team:** Photographer, AudioSpecialist, GraphicDesigner, DecoratingArchitecht,
- **Task**: A piece of work that is assigned to users in EventSystem.

2. Identified actors of the problem

- HumanResources
- FinancialManager
- AdministrationManger
- ProductionManager
- Photography, Music
- GraphicDesign
- Decoration
- NetworkSupport
- ServiceManager
- TopChef
- SeniorWaitress
- VicePresident
- Marketing
- CustomerServiceOfficers
- SeniorCustomerServiceOfficer,

3. Identified scenarios of the problem

Scenario name: SeniorCustomerServiceBooksFirstBusinessMeeting
Actors: Sarah:CustomerServiceOfficer, Janet:SeniorCustomerServiceOfficer, Alice:FinancialManager, Mark:AdministrationManager
Flow of events: 1. Sarah gets a call from a client who’s representing a company that wants to throw an annual party. Sarah opens a request for event planning form and fills it out according with the client's request such as when the party is to be held, what food is to be served and what the budget is. After the call and the form has been filled out Sarah proceeds to submit. 2. Janet receives a notification that there is a new request for event planning that needs her attention. She proceeds to open the form for the request and sees that everything is in order, so she approves the request. 3. Alice is alerted with a small pop-up window in the corner that she has a new task to review. She clicks on the window and the request for event planning opens up, now with an extra field marked for financial feedback. Alice fills out that field based on the client’s initial budget and what is needed for the annual party. She then continues by submitting the form. 4. Mike, the AdministrationManager, receives a notification that Alice has submitted a request for event planning with her financial feedback. Mike review the form submitted and acknowledges the report and forwards it to Janet by accepting the request.

5. Again Janet receives a notification that there is a new request that needs her attention, now from Mike. Since the request has been accepted she contacts the client and after discussing the budget that was based on the financial advice from Alice, the client accepts. Then they schedule a business meeting the following Monday.

Scenario name: AdminstrationApprovesGeneratedReport

Actors: Mark:AdministrationManager, Sam:CustomerService, Charlie:VicePresident, Alice:FinancialManager

Flow of events:

1. The day after the a Gaming conference event, Mark receives notification in EventSystem after logging in that he should create a new, empty report about the just held event. The notification displays a button that can be clicked that opens up a new empty report.
2. Mark writes his notes about what went well and what didn’t in a standard form for future lessons, then creates a task for Sam in the EventReport to gather all figures and numbers related to the event.
3. Sam receives a notification that the reports need some charts to display how many guests attended, when they arrived and how much they paid for their tickets.
4. Sam sends a request to Alice to access information about what he needs for the report. Alice receives his message and decides that it is no problem since the information given is not sensitive in any way. After a short while, Sam receives a file with all the necessary information.
5. Sam looks at the data but knows that it’s not of any use unless it is put together in a human friendly and understandable way. He knows that Charlie is in charge of making charts and is good with statistics.
6. Charlie, if not busy, gladly accepts the request and puts together a chart that fits the company's standard. He sends Sam the charts back.
7. Sam assembles the images, charts and texts to make the EventReport look as authentic as it can. After stylising the report to his imagination, he sends the report back to Mark and asks if he would like to add anything to it.
8. Mark is quite happy with the report since it gives him insight about the gaming industry and how successful the event really was. He now saves the document to the cloud so other employees of the company might want to read for future reference for similar events.

Scenario name: SendTaskToDecorationTeam

Actors: Jack:ProductionManager, Magy:DecoratingArchitect

Flow of events:

1. When Jack has finished scheduling staff for a government dinner party he logs into EventSystem and goes to the event home page. On the page he can see that no sub-teams have been sent tasks. He clicks on ‘Create tasks for sub-teams’, A form opens where Jack can choose from a drop-down menu that the task is being sent to the decoration team.
2. The form expands with the necessary fields and Jack can see that the information relevant to decoration in the client’s request form has been automatically filled in. Jack adds a comment and submits the form.
3. Magy, which had been scheduled to the event previously, sees a pop-up window on her screen that she has received a task.
4. She clicks on the window and the task opens up with the decoration requirements for the party. She checks the stock to see what is already available but sees that she has to buy more banners and flower pots, since a lot of them went missing wedding that was held the previous weekend.
5. She chooses to edit the task with how she is going to decorate, with what and approximately how long it will take. She also adds in what she will need to buy specifically for this dinner party, e.g. the banners and flower pots. Then she chooses to submit the edited task.
6. EventSystem notifies Jack that Magy has submitted her edited task so he opens the task and reviews it.

Scenario name: ScheduleStaffForWedding

Actors: Janet:SeniorCustomerServiceOfficers, Simon: HRTeam, Jack:ProductionManager, Tobias, Magdalena: Photographers

Flow of events:

1. After the FirstBusinessMeeting with the clients, Jack needs to start schedule staff for the wedding. After reading through the notes that the client provided he notices that the client needs a photographer for the wedding.
2. Jack sends a message to Tobias and Magdalena that he needs them to document the reception and the party after, but both of them are busy during the wedding date due to other events.
3. Jack knows that photos of the wedding are important so he sends a request to the Simon, asking for additional resources in the form of a freelancing photographer that can be hired for the event.
4. Simon looks at the request and starts searching for freelance photographers either on the internet, a photographing agency or someone he knows..
5. Jack sets himself in contact with a freelance photographer that has worked for the SEP before and did a good job.
6. The freelancing photographer agrees to the given task and is scheduled for the wedding on a given date.
7. Simon sends Jack a message that an photographer has agreed to the event as well as information about the photographer and his contact details.

4. Identified use cases for the problem.

1. Use Case Description

Name: ManageEventRequest

Initiating actor(s): CustomerService, SeniorCustomerService, FinancialManager, AdministrationManager

Entry Conditions:

1. CustomerService has successfully logged into the EventSystem
2. CustomerService is contacted by a client with a request for event planning.

Exit Conditions:

1. The SeniorCustomerService reject the request.
2. The AdministrationManager rejects the request based on his expectations or because the FinancialManager feedback shows the event as infeasible
3. The SeniorCustomerService schedules a business meeting with a client.

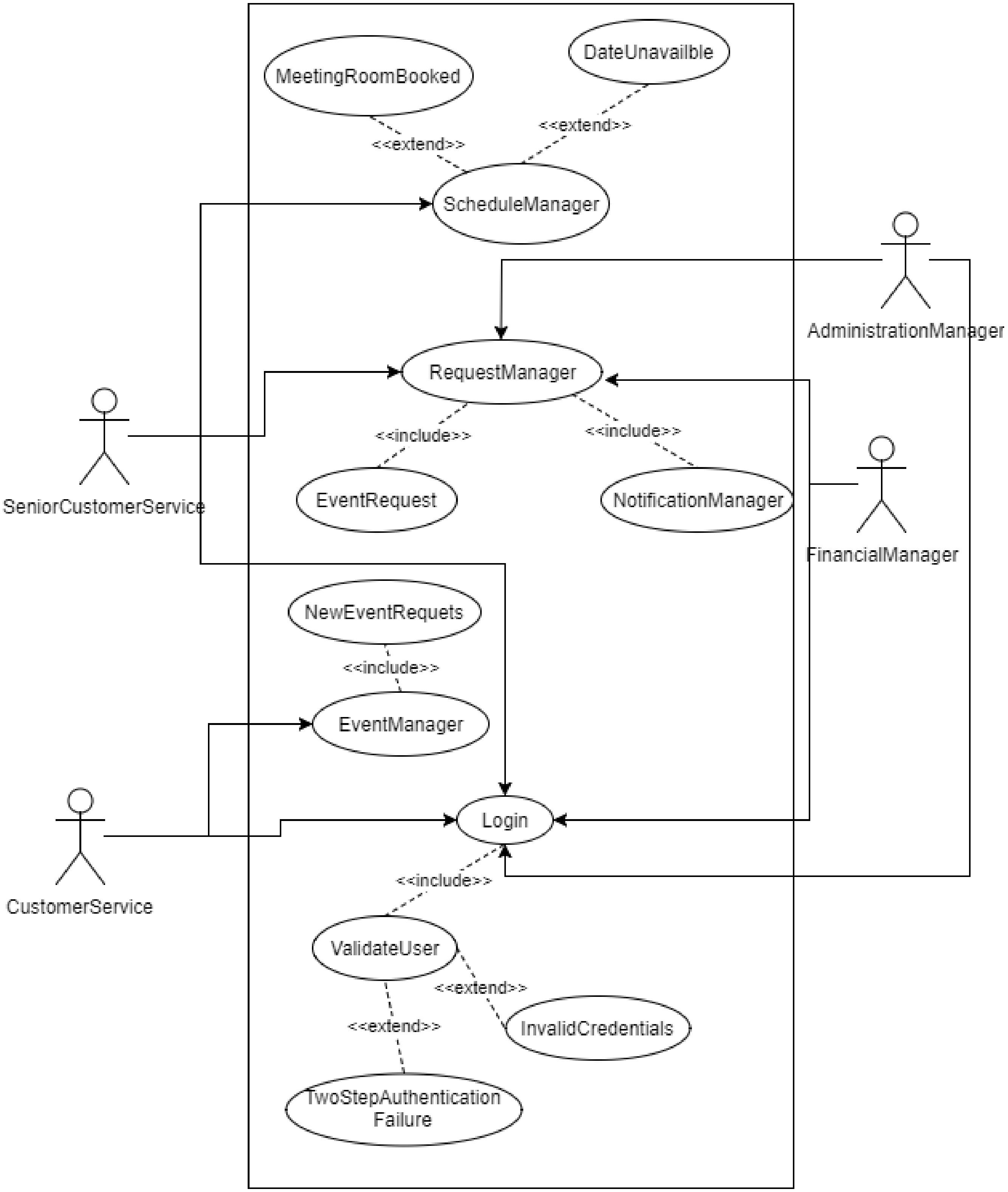
Quality Conditions: EventSystem should be available and functioning without unexpected interruptions.

Event Flow:

1. When CustomerService is contacted by a client with a request for event planning. CustomerService clicks on ‘Create New Request’.
    2. EventSystem displays an empty ‘Request for Event Planning’ form.
3. CustomerService fills in the form according to the clients wishes, such as name, event type, date, expected number of attendees, expected budget and the client’s preferences. Then CustomerService submits the form.
    4. EventSystem uploads the form, creates a new EventRequest, and sends it to the SeniorCustomerService while also notifying him that he has a new request form to review.
5. The SeniorCustomerService sees the notification window and clicks on it.
    6. EventSystem shows the request on the screen.
7. The SeniorCustomerService reviews the request. He has the option of rejecting or accepting the request. He chooses to accept the request.
    8. EventSystem sends the form to the FinancialManager and notifies him that he has a new request to review.
9. The FinancialManager clicks on the notification window
    10. EventSystem displays the request form with an additional feedback field for the FinancialManager to fill.

11. The FinancialManager reviews the form and writes in the form his feedback based on the original client’s budget idea and how large the FinancialManager thinks the budget needs to be. He then continues by submitting the form.
    12. EventSystem saves the edited form and then forwards the it to the AdministrationManager while also sending him a notification.
13. The AdministrationManager can see in his EventLog that he has a pending EventRequest that needs his attention. He clicks on it.
    14. EventSystem displays the EventRequest
15. The AdministrationManager reviews the request, he can either reject the approve the request. Based on his expectations and the feedback from the FinancialManager he chooses to approve.
    16. EventSystem send the approved EventRequest to the SeniorCustomerService with an appropriate notification.
17. The SeniorCustomerService opens the approved EventRequest and contacts the client to schedule a FirstBusinessMeeting. She clicks on ‘View Schedule’.
    18. EventSystem shows a calendar with the ProductionManager’s, FinancialManager’s, SeniorCustomerService’s and ServiceDepartmentManager schedule.
19. By using that information the SeniorCustomerService makes an appointment for the business meeting with the client that suits everyone. In the calendar menu the SeniorCustomerService chooses to create a new business meeting at the scheduled time and who are supposed to attend.
    20. EventSystem creates a new business meeting in the schedule of those who are supposed to attend and notifies them.

1. Use Case Diagram



2. Use Case Description

Name: GenerateEventReport

Participating actor(s): AdministrationManager, CustomerService, VicePresident, FinancialManager

Entry Conditions:

- The AdministrationManager is successfully logged into EventSystem AND
- It is the next weekday after a finished event AND
- EventSystem has sent a notification to the AdministrationManager that there is a pending EventReport to be filled.

Exit Conditions:

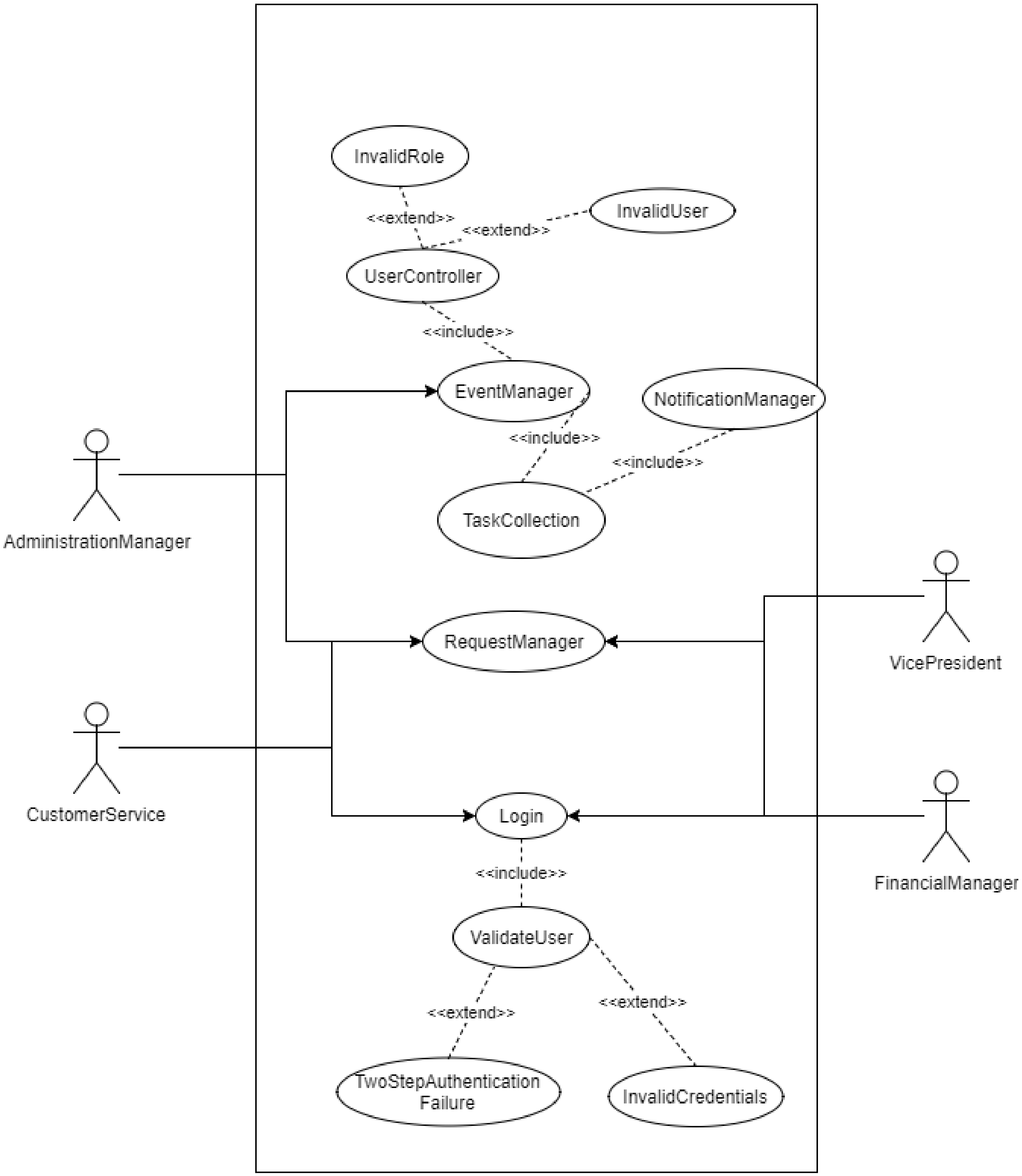
- EventReport is successfully saved in companies database. OR
- FinancialManager denies the request since the data about the event is sensitive

Quality Conditions: EventSystem should be available and functioning without unexpected interruptions. EventSystem must also successfully save the report.

Event Flow:

1. The AdministrationManager sees the notification that there is an empty EventReport that needs to be done.
2. AdministrationManager then creates a task related to Event, where he writes a description and a comment and submits it.
    3. EventSystem notifies the CustomerService about the pending EventReport.
4. CustomerService sends request to the FinancialManager for EventData.
    5. EventSystem displays request for the FinancialManager
6. The FinancialManager receives request to gather EventData for event and/or client. He then chooses to approve.
7. The FinancialManager gathers EventData regarding event from database and sends to the CustomerService.
    8. EventSystem sends EventData to CustomerService and displays it.
9. CustomerService sends approved data to the VicePresident
    10. EventSystem sends EventData to VicePresident and displays it.
11. The VicePresident generates charts and statistics from data.
12. He then confirms the chart and sends to the CustomerService
    13. EventSystem sends formatted EventData to CustomerService.
14. The CustomerService adds the statistics and charts to the EventReport
15. The AdministrationManager saves the EventReport to EventSystem.
    16. EventSystem saves EventReport to Fileserver.

2. Use Case Diagram



3. Use Case Description

Name: CreateTasksForSubTeams

Participating actor(s): ProductionManager, Sub-team, FinancialManager.

Entry Conditions:

- ProductionManager is logged into the EventSystem AND
- Staff has been successfully assigned staff.

Exit Conditions:
The ProductionManager sets the event's status to 'pending'

Quality Conditions: EventSystem should be available and functioning without unexpected interruptions.

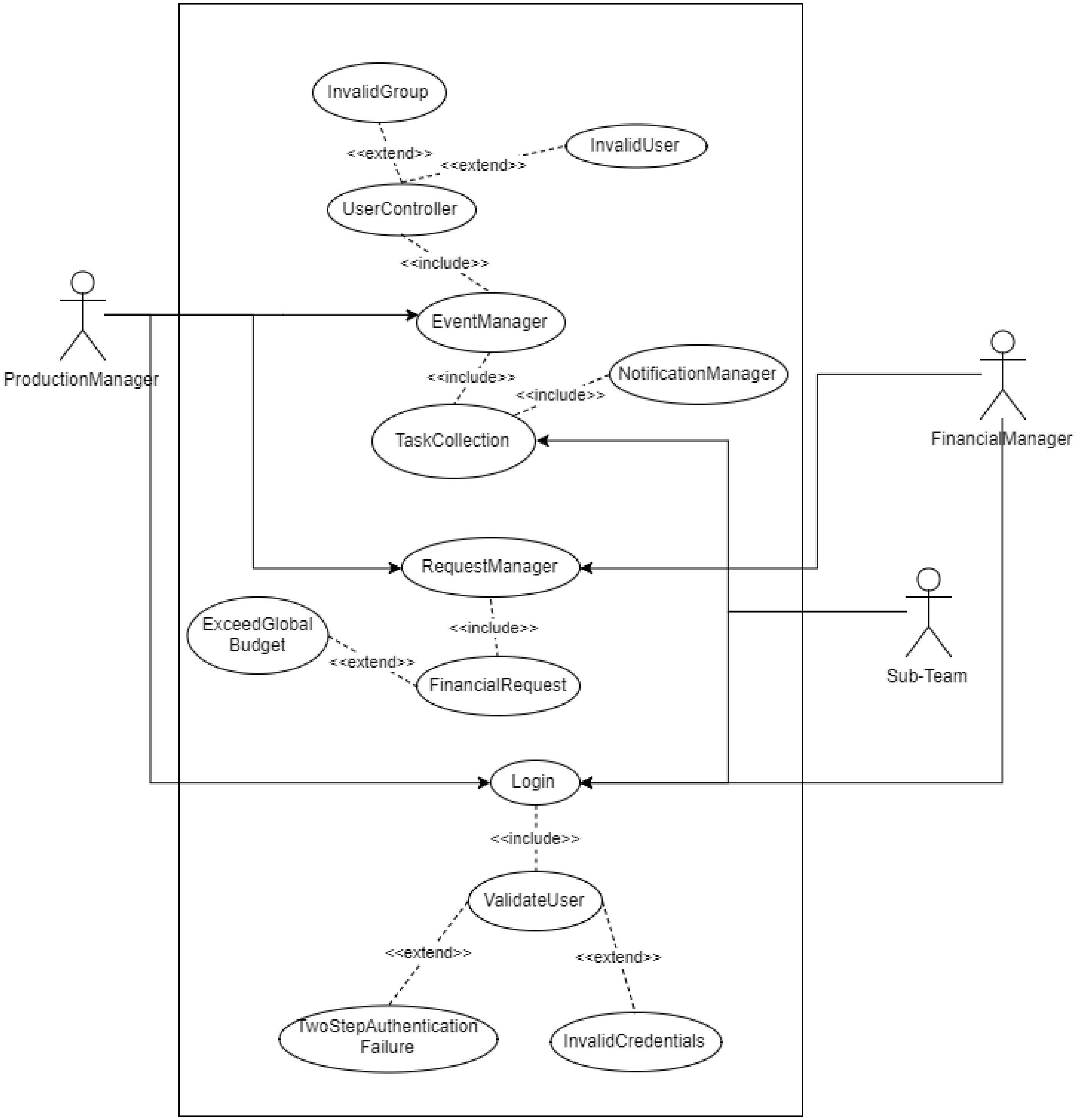
Event Flow:

1. The ProductionManager opens the list of upcoming events that need his attention, he can see that one event has staff that have not been given tasks, he chooses to view the EventHomePage for the event.
    2. EventSystem displays the EventHomePage, there is a list there with the assigned sub-team.
3. Next the ProductionManager clicks on 'Create task' that is displayed next to the sub-teams.
    4. EventSystem displays a form for the sub-team which has been automatically filled with relevant information from the 'Client Request Detail' form that was filled during the business meeting.
5. The ProductionManager sees the task form and edits it with a comment and other information that he sees that haven't been automatically filled but are crucial. He then clicks on 'Send'. He then continues to do this for every sub-team that has not been assigned a task.
    6. EventSystem saves the form and sends it to the person that was assigned to the sub-team.
7. The sub-team receives a notification that they have received a task and open it.
    8. EventSystem shows the task form on the screen.
9. The sub-team plans accordingly and edits the task with information about how long the task will take, how it will be done, what is needed and extra expenses that have to be taken into account. The sub-team then continues by sending the edited task back to the ProductionManager.
    10. EventSystem saves the edited task and sends it to the ProductionManager
11. The ProductionManager views the task sent from the sub-team, he sees that he will need to ask for a larger budget so he chooses 'Create Financial Request'
    12. EventSystem opens up the financial request form.
13. The ProductionManager fills in the details regarding which department the funds are

for, the amount asked for and reason for the request. He then submits the form.

14. EventSystem sends the request to the FinancialManager.
15. The FinancialManager reviews the request and has the option of denying the request or approve it, and he does the latter.
16. EventSystem sends a notification to the ProductionManager that his financial request has been accepted.
17. The ProductionManager can then continue by changing the event’s status from ‘pending’ to ‘in progress’.

3. Use Case Diagram



4. Use Case Description

Name: ScheduleStaffForEvent

Participating actor(s):, ProductionManager, HumanResources

Entry Conditions:
The first business meeting is finished and the client request detail has been filled.
The ProductionManager is logged into EventSystem.

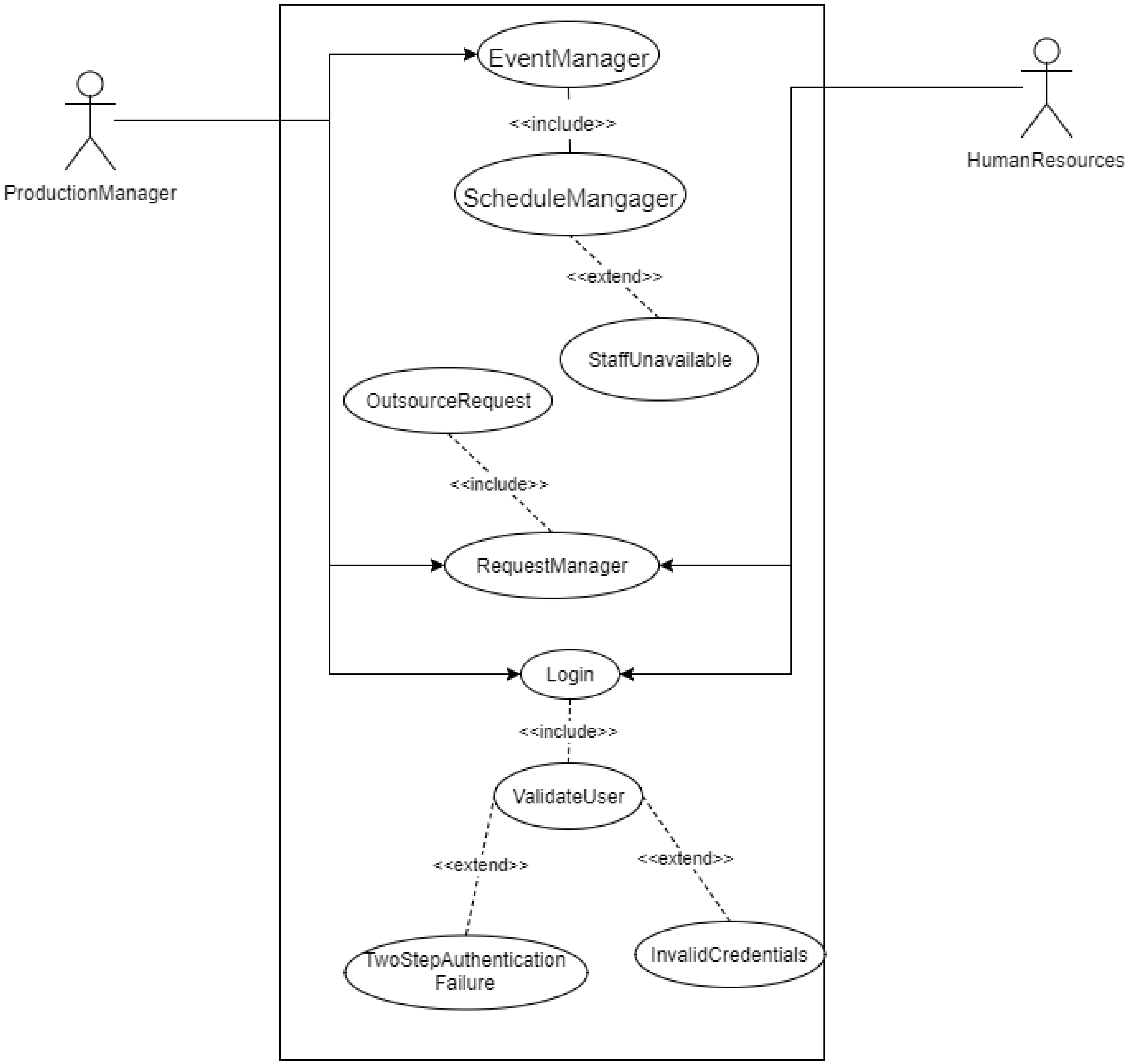
Exit Conditions:
The ProductionManager submits the staff scheduling for the event.

Quality Conditions: EventSystem should be available and functioning without unexpected interruptions.

Event Flow:

1. The ProductionManager chooses to view the event.
    2. EventSystem shows a list of upcoming events.
3. The ProductionManager chooses to schedule staff for one of the events
    4. EventSystem opens up a form with the events information and what positions need to be filled.
5. For each position that needs to be filled the production manager views a relevant employees schedule to find someone that’s free
6. If the employees are available they are scheduled
7. If not the ProductionManager contacts HumanResources to request to outsource the positions, either by hiring a long term employee or a freelancer.
8. HumanResources opens up a request form from the ProductionManager reviews it and accepts
9. The ProductionManager contacts a freelancer agency and hires an additional help for the event
10. The ProductionManager finishes the event scheduling form and submits it.

4. Use Case Diagram



5. Refinement of the use case diagrams

(adding alternative flows, and exceptions, use case extension and inclusion, …)

Extra refinements, flows and extensions are included in the UML diagrams presented in chapter 4.

6.Description of non-functional requirements.

Category	**Nonfunctional Requirements**
Implementation	Back-end must be developed with Windows Server IIS 7.5 for compatibility with other systems
Reliability	Role delegation and authorization should be configured when users login.
Reliability	EventSystem should support Two-Step verification and Electronic ID authentication.
Usability	EventSystem must allow authenticated users to download files from Filesystem on the device they are currently using.
Implementation	The Filesystem that hosts data regarding Tasks, EventReports and other related things of the EventSystem should be hosted in a cloud based solution.
Implementation	The front-end of the EventSystem should be browser based with HTTPS, CSS and Javascript so it can be accessed from all devices that support browsers.
Interface	The EventSystem must support a communication module for messaging, both for one-to-one and many-to-one.
Implementation	The architecture should be developed as a thin-client, Three tier architecture layer.
Performance	EventReports that are 5 years old or older should be migrated at the beginning of every year to another server that works as a archive to ease load balance on live server.
Usability	After authentication, a newly created Tasks that are assigned to the user should be visible as a red notification in the top right corner.
Interface	Text messages send in a chatbox should all be encrypted to ISO 27001 standard since users might be talking about sensitive information.
Implementation	All input to database should be written with parameters to block against possible SQL injections and multiple database requests should be made with a single transaction.