 **MINISTRY OF EDUCATION AND TRAINING**

**FPT UNIVERSITY**

Capstone Project Document

**Cinema Ticket**

|  |  |
| --- | --- |
| **Group 2** | |
| **Group members** | Lê Công Danh– Team Leader – SE61769  Lưu Lạc – Team Member – SE62056  Hồ Sĩ Hoàng – Team Member – SE61397 |
| **Supervisor** | Ms. Nguyễn Thị Cẩm Hương |
| **Ext. Supervisor** | N/A |
| **Capstone Project code** | OCBS |

- Ho Chi Minh City, 08/2017 –

***ACKNOWLEDGEMENTS***

We would like to take this opportunity to express our profound gratitude towards all the people who have contributed to this project: Our teachers for their advices and participation in the final review, our friends for their valuable technical support.

We would also like to express our appreciation to Ms. Nguyễn Thị Cẩm Hương, our research supervisor for her professional guidance and the useful, constructive recommendations throughout the course of this project.

Finally, we would like to thank our Families and Friends, for their unconditional support and tolerance during the time of doing the project.

# Table of Contents

[A. Introduction 15](#_Toc521515481)

[1. Project Overview 15](#_Toc521515482)

[2. Introduction 15](#_Toc521515483)

[3. Current Situation 15](#_Toc521515484)

[4. Problem Definition 16](#_Toc521515485)

[5. Proposed Solution 16](#_Toc521515486)

[**5.1. Featured functions** 16](#_Toc521515487)

[**5.2. Possible drawbacks** 17](#_Toc521515488)

[6. Functional Requirements 17](#_Toc521515489)

[7. Roles and Responsibilities 18](#_Toc521515490)

[B. Software Project Management Plan 19](#_Toc521515491)

[1. Problem Definition 19](#_Toc521515492)

[**1.1.** Name of this Capstone Project 19](#_Toc521515493)

[**1.2.** Problem Abstract 19](#_Toc521515494)

[**1.3.** Project Overview 19](#_Toc521515495)

[1.3.1. Current diagnostic diseases 19](#_Toc521515496)

[1.3.2. The Proposed System 19](#_Toc521515497)

[1.3.2.1. Web Application 20](#_Toc521515498)

[1.3.2.2. Mobile application 20](#_Toc521515499)

[1.3.3. Boundaries of the System 20](#_Toc521515500)

[1.3.4. Future plans 21](#_Toc521515501)

[1.3.5. Development Environment 21](#_Toc521515502)

[1.3.5.1. Hardware requirement 21](#_Toc521515503)

[1.3.5.2. Software requirement 21](#_Toc521515504)

[2. Project organization 22](#_Toc521515505)

[**2.1.** Software Project Model 22](#_Toc521515506)

[**2.2.** Roles and Responsibilities 23](#_Toc521515507)

[**2.3.** Tools and Techniques 23](#_Toc521515508)

[3. Project Management Plan 24](#_Toc521515509)

[**3.1.** Product Backlog 24](#_Toc521515510)

[**3.2.** Sprint Backlog 25](#_Toc521515511)

[3.2.1. Sprint 1 (14/5/2018 -27/5/2018) Project Initiation 25](#_Toc521515512)

[3.2.1.1. Goal 25](#_Toc521515513)

[3.2.1.2. Development 26](#_Toc521515514)

[3.2.2. Sprint 2 (28/5/2018 - 3/6/2018) Software requirement and basic features 26](#_Toc521515515)

[3.2.2.1. Goal 26](#_Toc521515516)

[3.2.2.2. Development 26](#_Toc521515517)

[3.2.3. Sprint 3 (4/6/2018 - 17/6/2018) Software Design Description and main features 27](#_Toc521515518)

[3.2.3.1. Goal 27](#_Toc521515519)

[3.2.3.2. Development 27](#_Toc521515520)

[3.2.4. Sprint 4 (18/6/2018 – 8/7/2018) Finish up coding 28](#_Toc521515521)

[3.2.4.1. Goal 28](#_Toc521515522)

[3.2.4.2. Development 28](#_Toc521515523)

[3.2.5. Sprint 5 (8/7/2018 – 22/7/2018) Software Test Document and Quality Assurance 29](#_Toc521515524)

[3.2.5.1. Goal 29](#_Toc521515525)

[3.2.5.2. Development 29](#_Toc521515526)

[3.2.6. Sprint 6 (22/7/2018 – 5/8/2018) Software User’s Manual 29](#_Toc521515527)

[3.2.6.1. Goal 29](#_Toc521515528)

[3.2.6.2. Development 29](#_Toc521515529)

[**3.3.** Deliverables 30](#_Toc521515530)

[**3.4.** All Meeting Minutes 30](#_Toc521515531)

[4. Coding Convention 30](#_Toc521515532)

[C. Software Requirement Specifications 31](#_Toc521515533)

[1. User Requirement Specification 31](#_Toc521515534)

[**1.1.** Admin Requirement 31](#_Toc521515535)

[**1.2.** Partner Requirement 32](#_Toc521515536)

[**1.3.** Cinema Manager Requirement 32](#_Toc521515537)

[**1.4.** User Requirement 32](#_Toc521515538)

[**1.5.** Customer Requirement 32](#_Toc521515539)

[2. Software Requirement Specification 32](#_Toc521515540)

[**2.1.** External Interface Requirement 33](#_Toc521515541)

[2.1.1. User Interface 33](#_Toc521515542)

[2.1.2. Hardware Interface 33](#_Toc521515543)

[2.1.3. Software Interface 33](#_Toc521515544)

[2.1.4. Communication Protocol 33](#_Toc521515545)

[**2.2.** System Overview Use Case 33](#_Toc521515546)

[**2.3.** List of Use Case 34](#_Toc521515547)

[2.3.1. <Admin> Overview Use Case 34](#_Toc521515548)

[2.3.1.1. <Admin> View All Group Cinema (UC\_AD002) 35](#_Toc521515549)

[2.3.1.2. <Admin> Add new Group Cinema (UC\_AD003) 36](#_Toc521515550)

[2.3.1.3. <Admin> Edit group cinema information account (UC\_AD004) 39](#_Toc521515551)

[2.3.1.4. <Admin> View all Partner (UC\_AD05) 41](#_Toc521515552)

[2.3.1.5. <Admin> Create Account for Partner (UC\_AD006) 42](#_Toc521515553)

[2.3.1.6. <Admin> Edit partner account (UC\_AD007) 44](#_Toc521515554)

[2.3.1.7. <Admin> Delete partner account (UC\_AD007) 45](#_Toc521515555)

[2.3.1.8. <Admin> View List Film (UC\_AD009) 47](#_Toc521515556)

[2.3.1.9. <Admin> Add Film (UC\_AD010) 48](#_Toc521515557)

[2.3.1.10. <Admin> Edit Film (UC\_AD011) 50](#_Toc521515558)

[2.3.1.11. <Admin> Crawl Film Data (UC\_AD012) 53](#_Toc521515559)

[2.3.2. <Partner> Overview Use Case 54](#_Toc521515560)

[2.3.2.1. <Partner>View All Cinema Manager (UC\_P002) 54](#_Toc521515561)

[2.3.2.2. <Partner> Add Cinema Manager Account (UC\_P003) 56](#_Toc521515562)

[2.3.2.3. <Partner>Delete Cinema Manager Account (UC\_P004) 58](#_Toc521515563)

[2.3.2.4. <Partner> Edit Cinema Manager Account (UC\_P005) 59](#_Toc521515564)

[2.3.2.5. <Partner> View all cinema (UC\_P006) 61](#_Toc521515565)

[2.3.2.6. <Partner> Arrange seat (UC\_P007) 62](#_Toc521515566)

[2.3.2.7. <Partner> Add Room for Cinema (UC\_P008) 63](#_Toc521515567)

[2.3.2.8. <Partner> Add New Cinema (UC\_P009) 65](#_Toc521515568)

[2.3.2.9. <Partner> Edit cinema (UC\_P010) 67](#_Toc521515569)

[2.3.2.10. <Partner> View Room List (UC\_P011) 69](#_Toc521515570)

[2.3.2.11. <Partner> View Room Detail (UC\_P012) 71](#_Toc521515571)

[2.3.3. <User> Overview Use Case 72](#_Toc521515572)

[2.3.3.1. <User> View Profile (UC\_U002) 72](#_Toc521515573)

[2.3.3.2. <User> View Booking History (UC\_U004) 74](#_Toc521515574)

[2.3.4. <Customer> Overview Use Case 75](#_Toc521515575)

[2.3.4.1. <Customer> Register (UC\_C001) 75](#_Toc521515576)

[2.3.4.2. <Customer> Find Schedule (UC\_C002) 77](#_Toc521515577)

[2.3.4.3. <Customer>Choose type of seat and quality (UC\_C003) 79](#_Toc521515578)

[2.3.4.4. <Customer> Choose Seat (UC\_C004) 81](#_Toc521515579)

[2.3.4.5. <Customer> Checkout (UC\_C005) 82](#_Toc521515580)

[2.3.4.6. <Customer> Resell ticket (UC\_C006) 84](#_Toc521515581)

[2.3.4.7. <Customer> Change ticket (UC\_C007) 87](#_Toc521515582)

[2.3.5. <Cinema manager> Overview User Case 88](#_Toc521515583)

[2.3.5.1. <Cinema manager > Basic add movie schedule (UC\_M002) 89](#_Toc521515584)

[2.3.5.2. <Cinema manager > Advantage schedule (UC\_M003) 90](#_Toc521515585)

[2.3.5.3. <Cinema manager > View dashboard (UC\_M004) 93](#_Toc521515586)

[2.3.6. <System> Overview Use Case 94](#_Toc521515587)

[2.3.6.1. <System> Notification about schedule (UC\_S002) 94](#_Toc521515588)

[2.3.6.2. <System> suggest film for user (UC\_S003) 96](#_Toc521515589)

[3. Software Requirement Specification 97](#_Toc521515590)

[**3.1.** Usability 97](#_Toc521515591)

[**3.2.** Reliability 97](#_Toc521515592)

[**3.3.** Availability 97](#_Toc521515593)

[**3.4.** Security. 97](#_Toc521515594)

[**3.5.** Maintainability 97](#_Toc521515595)

[**3.6.** Portability 97](#_Toc521515596)

[**3.7.** Performance 97](#_Toc521515597)

[4. Conceptual Diagram 98](#_Toc521515598)

[D. Software Design Description 100](#_Toc521515599)

[1. Design Overview 100](#_Toc521515600)

[2. System Architectural Design 100](#_Toc521515601)

[**2.1.** Web Application Architecture Description 100](#_Toc521515602)

[**2.2.** Web Service Application Architecture Description 102](#_Toc521515603)

[**2.3.** Mobile Application Architecture Description 102](#_Toc521515604)

[3. Component Diagram 103](#_Toc521515605)

[4. Detailed Description 104](#_Toc521515606)

[**4.1.** Class Diagram 104](#_Toc521515607)

[**4.2.** Class Diagram Explanation 105](#_Toc521515608)

[**4.3.** Interaction Diagram 106](#_Toc521515609)

[4.3.1. Sequence Diagram 106](#_Toc521515610)

[4.3.1.1. <User> Booking ticket 106](#_Toc521515611)

[4.3.1.2. <User> Resell ticket 106](#_Toc521515612)

[4.3.1.3. <User> Change ticket 107](#_Toc521515613)

[4.3.1.4. <User> User register 107](#_Toc521515614)

[4.3.1.5. <User> User login 107](#_Toc521515615)

[4.3.1.6. <User> User logout 107](#_Toc521515616)

[4.3.1.7. <User> Edit profile 108](#_Toc521515617)

[4.3.1.8. <User> View booking ticket history 108](#_Toc521515618)

[4.3.1.9. <Partner> Partner login 108](#_Toc521515619)

[4.3.1.10. <Partner> Partner create new cinema 109](#_Toc521515620)

[4.3.1.11. <Partner> Partner add room for cinema 109](#_Toc521515621)

[4.3.1.12. <Partner> Partner edit cinema information 109](#_Toc521515622)

[4.3.1.13. <Partner> Partner add cinema manager account 109](#_Toc521515623)

[4.3.1.14. <Partner> Partner edit cinema manager account information 110](#_Toc521515624)

[4.3.1.15. <Partner> Partner delete cinema manager account 110](#_Toc521515625)

[4.3.1.16. <Partner> Partner view report 110](#_Toc521515626)

[4.3.1.17. <Admin> Admin login 110](#_Toc521515627)

[4.3.1.18. < Admin > Admin add film 111](#_Toc521515628)

[4.3.1.19. < Admin > Admin edit film information 111](#_Toc521515629)

[4.3.1.20. < Admin > Admin add group cinema 111](#_Toc521515630)

[4.3.1.21. < Admin > Admin edit group cinema information 111](#_Toc521515631)

[4.3.1.22. < Admin > Admin add partner account 112](#_Toc521515632)

[4.3.1.23. < Admin > Admin edit partner account information 112](#_Toc521515633)

[4.3.1.24. < Admin > Admin delete partner account 112](#_Toc521515634)

[4.3.2. Activity Diagram 112](#_Toc521515635)

[4.3.2.1. Login 113](#_Toc521515636)

[4.3.2.2. Booking Ticket Web Application 113](#_Toc521515637)

[4.3.2.3. Booking Ticket Mobile Application 114](#_Toc521515638)

[5. Interface 115](#_Toc521515639)

[**5.1.** Web Application 115](#_Toc521515640)

[5.1.1. User 115](#_Toc521515641)

[5.1.1.1. Home Page 115](#_Toc521515642)

[5.1.1.2. Film Detail 116](#_Toc521515643)

[5.1.1.3. Choose Ticket 117](#_Toc521515644)

[5.1.1.4. Choose Seat 118](#_Toc521515645)

[5.1.2. Admin 119](#_Toc521515646)

[5.1.2.1. Manage Film 119](#_Toc521515647)

[5.1.2.2. Manage Partner 119](#_Toc521515648)

[5.1.2.3. Manage Group of Cinema 120](#_Toc521515649)

[5.1.3. Partner 120](#_Toc521515650)

[5.1.3.1. Manage Cinema 120](#_Toc521515651)

[5.1.3.2. Manage Employee 121](#_Toc521515652)

[5.1.4. Cinema Manager 121](#_Toc521515653)

[5.1.4.1. Manage Schedule 121](#_Toc521515654)

[**5.2.** Mobile Application 122](#_Toc521515655)

[5.2.1. Front Page 122](#_Toc521515656)

[5.2.2. Cinema Page 124](#_Toc521515657)

[5.2.3. User Page 126](#_Toc521515658)

[5.2.4. Purchase Ticket Page 129](#_Toc521515659)

[6. Database Design 132](#_Toc521515660)

[**6.1.** Entity Relationship Diagram (ERD) 132](#_Toc521515661)

[**6.2.** Data Dictionary 133](#_Toc521515662)

[6.2.1. User 133](#_Toc521515663)

[6.2.2. Admin 134](#_Toc521515664)

[6.2.3. Customer 134](#_Toc521515665)

[6.2.4. Partner Account 134](#_Toc521515666)

[6.2.5. Cinema Manager 135](#_Toc521515667)

[6.2.6. Film 135](#_Toc521515668)

[6.2.7. Movie schedule 136](#_Toc521515669)

[6.2.8. Show Time 136](#_Toc521515670)

[6.2.9. Room 136](#_Toc521515671)

[6.2.10. Digital Type 137](#_Toc521515672)

[6.2.11. Seat 137](#_Toc521515673)

[6.2.12. Type of Seat 137](#_Toc521515674)

[6.2.13. Group Cinema 138](#_Toc521515675)

[6.2.14. Cinema 138](#_Toc521515676)

[6.2.15. Ticket 138](#_Toc521515677)

[6.2.16. Booking Ticket 139](#_Toc521515678)

[7. Algorithms 139](#_Toc521515679)

[**7.1.** Display physical cinema room in 2D interface. 139](#_Toc521515680)

[7.1.1. Define problem. 139](#_Toc521515681)

[7.1.2. Solution. 139](#_Toc521515682)

[**7.2.** Select group of seat on room’s matrix. 140](#_Toc521515683)

[7.2.1. Define problem. 140](#_Toc521515684)

[7.2.2. Solution. 140](#_Toc521515685)

[**7.3.** Movie recommendations for users. 142](#_Toc521515686)

[7.3.1. Define problem. 142](#_Toc521515687)

[7.3.2. Solution. 142](#_Toc521515688)

[E. System Implementation & Test 143](#_Toc521515689)

[1. Introduction 143](#_Toc521515690)

[**1.1.** Overview 143](#_Toc521515691)

[**1.2.** Test Approach 143](#_Toc521515692)

[2. Database Relationship Diagram 143](#_Toc521515693)

[**2.1.** Physical Diagram 143](#_Toc521515694)

[**2.2.** Data Dictionary 143](#_Toc521515695)

[3. Implementation 147](#_Toc521515696)

[**3.1.** Backend Architecture Implementation 147](#_Toc521515697)

[**3.2.** Mobile Application Architecture Implementation 149](#_Toc521515698)

[**3.3.** Web Architecture Implementation 150](#_Toc521515699)

[**3.4.** Test Driven Development 151](#_Toc521515700)

[4. Test Plan 151](#_Toc521515701)

[**4.1.** Communication Diagram 151](#_Toc521515702)

[4.1.1. User 151](#_Toc521515703)

[User purchase ticket 151](#_Toc521515704)

[4.1.2. Admin 151](#_Toc521515705)

[4.1.2.1. Admin Manage Film 152](#_Toc521515706)

[4.1.2.2. Admin Manage Partner 152](#_Toc521515707)

[4.1.2.3. Admin Manage Group of Cinema 152](#_Toc521515708)

[4.1.3. Partner 152](#_Toc521515709)

[4.1.3.1. Partner Manage Cinema 153](#_Toc521515710)

[4.1.3.2. Partner Add Room 153](#_Toc521515711)

[4.1.4. Cinema Manager 153](#_Toc521515712)

[Cinema Manager Manage Schedule 153](#_Toc521515713)

[5. System Testing Test Case 153](#_Toc521515714)

[**5.1.** Web Application Test Cases 153](#_Toc521515715)

[5.1.1. <Customer> View Film Detail 153](#_Toc521515716)

[5.1.2. <Customer> Choose Tickets 154](#_Toc521515717)

[5.1.3. <Customer> Watch Film’s Trailer 157](#_Toc521515718)

[5.1.4. <Customer> Choose Seat 157](#_Toc521515719)

[5.1.5. <Customer> Purchase Tickets 160](#_Toc521515720)

[5.1.6. <Admin> Add Partner 163](#_Toc521515721)

[5.1.7. <Admin> Edit Partner 166](#_Toc521515722)

[5.1.8. <Admin> Delete Partner 169](#_Toc521515723)

[5.1.9. <Admin> Create Film 169](#_Toc521515724)

[5.1.10. <Admin> Edit Film 172](#_Toc521515725)

[5.1.11. <Admin> Add Group Cinema 175](#_Toc521515726)

[5.1.12. <Partner>Add Employee 175](#_Toc521515727)

[5.1.13. <Partner> Edit Employee 178](#_Toc521515728)

[5.1.14. <Partner> Delete Employee 179](#_Toc521515729)

[5.1.15. <Partner> Add Cinema 179](#_Toc521515730)

[5.1.16. <Partner> Edit Cinema 181](#_Toc521515731)

[5.1.17. <Partner> Add Room 183](#_Toc521515732)

[**5.2.** Mobile Application Test Cases 184](#_Toc521515733)

[5.2.1. <User> Login 184](#_Toc521515734)

[5.2.2. <User> View List Film 184](#_Toc521515735)

[5.2.3. <Authorized User>Purchase Ticket 185](#_Toc521515736)

[F. Software User’s Manual 187](#_Toc521515737)

[1. Installation Guide 187](#_Toc521515738)

[**1.1.** Setting up environment at server side 187](#_Toc521515739)

[1.1.1. Hardware Requirements 187](#_Toc521515740)

[1.1.2. Software Requirements 187](#_Toc521515741)

[1.1.3. Mobile Requirements 187](#_Toc521515742)

[2. User Guide 187](#_Toc521515743)

[**2.1.** Web application 187](#_Toc521515744)

[2.1.1. Resell ticket 187](#_Toc521515745)

[2.1.2. Booking ticket 188](#_Toc521515746)

[**2.2.** Mobile application 189](#_Toc521515747)

[2.2.1. Login 189](#_Toc521515748)

[2.2.2. Register 190](#_Toc521515749)

[2.2.3. Booking ticket 190](#_Toc521515750)

[2.2.4. Show all order 192](#_Toc521515751)

[2.2.5. Order detail 193](#_Toc521515752)

[2.2.6. Resell ticket 194](#_Toc521515753)

[2.2.7. Change ticket 195](#_Toc521515754)

[**2.3.** Admin web application 196](#_Toc521515755)

[2.3.1. Crawl film data and add film to system 196](#_Toc521515756)

[2.3.2. Add new film 197](#_Toc521515757)

[**2.4.** Manager web application 198](#_Toc521515758)

[2.4.1. Basic add new schedule 198](#_Toc521515759)

[2.4.2. Advantage Add Schedule 198](#_Toc521515760)

[G. Appendix 199](#_Toc521515761)

# List of Tables

[Table 1: Roles and Responsibilities 18](#_Toc521515762)

[*Table 2: Hardware requirement for Server* 21](#_Toc521515763)

[*Table 3: Hardware requirement for Web* 21](#_Toc521515764)

[*Table 4: Hardware requirement for Mobile* 21](#_Toc521515765)

[*Table 5: Software requirement* 22](#_Toc521515766)

[*Table 6: Roles and Responsibilities* 23](#_Toc521515767)

[*Table 7: Tools and Techniques* 24](#_Toc521515768)

[Table 8: Product Backlog 25](#_Toc521515769)

[*Table 9: Sprint 1 Development* 26](#_Toc521515770)

[*Table 10: Sprint 2 Development* 27](#_Toc521515771)

[*Table 11: Sprint 3 Development* 28](#_Toc521515772)

[*Table 12: Sprint 4 Development* 29](#_Toc521515773)

[*Table 13: Sprint 5 Development* 29](#_Toc521515774)

[*Table 14: Sprint 6 Development* 30](#_Toc521515775)

[*Table 15: Deliverables* 30](#_Toc521515776)

[Table 16: USE CASE - UC\_AD002 - <Admin> View all group cinema 36](#_Toc521515777)

[Table 17: USE CASE - UC\_AD003 - <Admin> Add new group cinema 39](#_Toc521515778)

[Table 18: USE CASE - UC\_AD004 - <Admin> Edit Group Cinema Information 41](#_Toc521515779)

[Table 19: USE CASE - UC\_AD005 - <Admin> View all Partner 42](#_Toc521515780)

[Table 20: USE CASE - UC\_AD006 - <Admin> Create account for Partner 44](#_Toc521515781)

[Table 21: USE CASE - UC\_AD007 - <Admin> Edit partner account 45](#_Toc521515782)

[Table 22: USE CASE - UC\_AD008 - <Admin> Delete partner account 47](#_Toc521515783)

[Table 23: USE CASE - UC\_AD009 - <Admin> View List Film 48](#_Toc521515784)

[Table 24: USE CASE - UC\_AD010 - <Admin> Add Film 50](#_Toc521515785)

[Table 25: USE CASE - UC\_AD011 - <Admin> Edit Film 53](#_Toc521515786)

[Table 26: <admin> Crawl Film Data (UC\_AD012) 54](#_Toc521515787)

[Table 27: USE CASE - UC\_P002 - <Partner> View all cinema manager account 56](#_Toc521515788)

[Table 28: USE CASE - UC\_P003 <Partner> Add Cinema Manager Account 58](#_Toc521515789)

[Table 29: USE CASE - UC\_P004 <Partner>Delete Cinema Manager Account 59](#_Toc521515790)

[Table 30: USE CASE - UC\_P005 <Partner> Edit Cinema Manager Account 61](#_Toc521515791)

[Table 31: USE CASE - UC\_P006 <Partner> View all cinema 62](#_Toc521515792)

[Table 32: USE CASE - UC\_P007 <Partner> Arrange seat 63](#_Toc521515793)

[Table 33: USE CASE - UC\_P008 <Partner> Add Room for Cinema 65](#_Toc521515794)

[Table 34: USE CASE - UC\_P009 <Partner> Add New Cinema 67](#_Toc521515795)

[Table 35: USE CASE - UC\_P010 <Partner> Edit cinema 69](#_Toc521515796)

[Table 36: USE CASE - UC\_P011 <Partner> View Room List 71](#_Toc521515797)

[Table 37: USE CASE - UC\_P012 <Partner> View Room Detail 72](#_Toc521515798)

[Table 38: USE CASE - UC\_U002 <User> View Profile 74](#_Toc521515799)

[Table 39:USE CASE - UC\_U004 <User> View Booking History 75](#_Toc521515800)

[Table 40: USE CASE - UC\_C001 <Customer> Register 77](#_Toc521515801)

[Table 41: USE CASE - UC\_C002 <Customer> Filter Ticket 79](#_Toc521515802)

[Table 42: USE CASE - UC\_C003 <Customer> Choose type of seat and quality 81](#_Toc521515803)

[Table 43: USE CASE - UC\_C004 <Customer> Choose Seat 82](#_Toc521515804)

[Table 44: USE CASE - UC\_C005 <Customer> Checkout 84](#_Toc521515805)

[Table 45: USE CASE - UC\_C006 <Customer> Resell ticket 87](#_Toc521515806)

[Table 46: USE CASE - UC\_C007 <Customer> Change ticket 88](#_Toc521515807)

[Table 47: USE CASE - UC\_M002 <Cinema manager > Basic add movie schedule 90](#_Toc521515808)

[Table 48: USE CASE - UC\_M003 <Cinema manager > Advantage schedule 93](#_Toc521515809)

[Table 49: USE CASE - UC\_M004 <Cinema manager > View dashboard 94](#_Toc521515810)

[Table 50: USE CASE - UC\_S002 <System> Notification about schedule 96](#_Toc521515811)

[Table 51: USE CASE - UC\_S003 <System> suggest film for user 97](#_Toc521515812)

[Table 52: Conceptual Diagram Data Dictionary 99](#_Toc521515813)

[Table 53: Component Dictionary 104](#_Toc521515814)

[Table 54: Class Diagram Dictionary 106](#_Toc521515815)

[Table 55: Mobile Application – Front Page - Interface 124](#_Toc521515816)

[Table 56: Mobile Application – Cinema Page - Interface 126](#_Toc521515817)

[Table 57: Mobile Application – User Page - Interface 129](#_Toc521515818)

[Table 58: Mobile Application – Purchase Ticket Page - Interface 132](#_Toc521515819)

[Table 59: Entities Description 133](#_Toc521515820)

[Table 60: User class attributes explanation 133](#_Toc521515821)

[Table 61: User class methods explanation 134](#_Toc521515822)

[Table 62: Admin class attributes explanation 134](#_Toc521515823)

[Table 63: Admin method methods explanation 134](#_Toc521515824)

[Table 64: Customer class attributes explanation 134](#_Toc521515825)

[Table 65: Customer method methods explanation 134](#_Toc521515826)

[Table 66: Partner class attributes explanation 135](#_Toc521515827)

[Table 67: Partner class methods explanation 135](#_Toc521515828)

[Table 68: Cinema Manager class attributes explanation 135](#_Toc521515829)

[Table 69: Cinema Manager class methods explanation 135](#_Toc521515830)

[Table 70: Film class attributes explanation 136](#_Toc521515831)

[Table 71: Film class methods explanation 136](#_Toc521515832)

[Table 72: Movie class attributes explanation 136](#_Toc521515833)

[Table 73: Show time class attributes 136](#_Toc521515834)

[Table 74: Show time class method explanation 136](#_Toc521515835)

[Table 75: Room class attributes explanation 136](#_Toc521515836)

[Table 76: Room class methods explanation 137](#_Toc521515837)

[Table 77: Digital type class attributes explanation 137](#_Toc521515838)

[Table 78: Digital Type class methods explanation 137](#_Toc521515839)

[Table 79: Seat class attributes explanation 137](#_Toc521515840)

[Table 80: Seat class methods explanation 137](#_Toc521515841)

[Table 81: Type of Seat class attributes explanation 137](#_Toc521515842)

[Table 82: Type of Seat class methods explanation 138](#_Toc521515843)

[Table 83: Group Cinema class attributes explanation 138](#_Toc521515844)

[Table 84: Group Cinema class methods explanation 138](#_Toc521515845)

[Table 85: Cinema class attributes explanation 138](#_Toc521515846)

[Table 86: Cinema class methods explanation 138](#_Toc521515847)

[Table 89: Ticket class attributes explanation 138](#_Toc521515848)

[Table 90: Ticket class methods explanation 139](#_Toc521515849)

[Table 91: Booking ticket class attributes explanation 139](#_Toc521515850)

[Table 92: Booking ticket class methods explanation 139](#_Toc521515851)

[Table 93: Entity Data Dictionary 144](#_Toc521515852)

[Table 94: Entity Data Dictionary 147](#_Toc521515853)

[Table 95: *Backend Architecture Implementation* 148](#_Toc521515854)

[Table 96: *Mobile Application Architecture Implementation* 149](#_Toc521515855)

[Table 97: *Web Architecture Implementation* 151](#_Toc521515856)

[*Table 98: Web application test case- <Customer> View Film Detail* 154](#_Toc521515857)

[*Table 99: Web application test case- <Customer> Choose Tickets* 157](#_Toc521515858)

[*Table 100: Web application test case- <Customer> Watch Film’s Trailer* 157](#_Toc521515859)

[*Table 101: Web application test case- <Customer> Choose Seat* 160](#_Toc521515860)

[*Table 102: Web application test case- <Customer> Purchase Ticket* 163](#_Toc521515861)

[*Table 103: Web application test case- <Admin> Add Partner* 166](#_Toc521515862)

[*Table 104: Web application test case- <Admin> Edit Partner* 169](#_Toc521515863)

[*Table 105: Web application test case- <Admin> Delete Partner* 169](#_Toc521515864)

[*Table 106: Web application test case- <Admin> Create Film* 172](#_Toc521515865)

[*Table 107: Web application test case- <Admin> Edit Film* 175](#_Toc521515866)

[*Table 108: Web application test case- <Admin> Add Group Cinema* 175](#_Toc521515867)

[*Table 109: Web application test case- <Partner> Add Employee* 178](#_Toc521515868)

[*Table 110: Web application test case- <Partner> Edit Employee* 179](#_Toc521515869)

[*Table 111: Web application test case- <Partner> Delete Employee* 179](#_Toc521515870)

[*Table 112: Web application test case- <Partner> Add Cinema* 181](#_Toc521515871)

[*Table 113: Web application test case- <Partner> Edit Cinema* 183](#_Toc521515872)

[*Table 114: Web application test case- <Partner> Add Room* 184](#_Toc521515873)

[Table 115: *Mobile application test case- <User> login* 184](#_Toc521515874)

[Table 116: *Mobile application test case- <User> View List Film* 185](#_Toc521515875)

[Table 117: *Mobile application test case- <User> Purchase Ticket* 186](#_Toc521515876)

[*Table 118: Installation Guide - Hardware Requirements* 187](#_Toc521515877)

[*Table 119: Installation Guide - Software Requirement* 187](#_Toc521515878)

[*Table 120: Installation Guide - Mobile Requirement* 187](#_Toc521515879)

[Table 121: User Guide - Web application – Resell ticket 188](#_Toc521515880)

[Table 122: User Guide - Web application –Booking ticket 189](#_Toc521515881)

[Table 123: User Guide - Mobile application – Login 190](#_Toc521515882)

[Table 124: User Guide - Mobile application – Register 190](#_Toc521515883)

[Table 125: User Guide - Mobile application – Booking ticket 192](#_Toc521515884)

[Table 126: User Guide - Mobile application – Show all order 193](#_Toc521515885)

[Table 127: User Guide - Mobile application – Order detail 194](#_Toc521515886)

[Table 128: User Guide - Mobile application – Resell ticket 195](#_Toc521515887)

[Table 129: User Guide - Mobile application – Change ticket 195](#_Toc521515888)

[Table 130: User Guide – Admin web application – Crawl film data and add film to system 197](#_Toc521515889)

[Table 131: User Guide – Admin web application – Add new film 198](#_Toc521515890)

[Table 132: User Guide – Manager web application – Basic add new schedule 198](#_Toc521515891)

[Table 133: User Guide – Manager web application – Advantage Add Schedule 199](#_Toc521515892)

**List of Figures**

[*Figure 1: Scrum Process* 22](#_Toc521515893)

[Figure 2: System Overview Use Case 34](#_Toc521515894)

[Figure 3: <Admin> Overview Use Case 35](#_Toc521515895)

[Figure 4: <Admin> View All Group Cinema (UC\_AD002) 35](#_Toc521515896)

[Figure 5: <Admin> Add new group cinema (UC\_AD003) 37](#_Toc521515897)

[Figure 6: <Admin> Edit group cinema information(UC\_AD04) 39](#_Toc521515898)

[Figure 7: <Admin> View All Partner (UC\_AD005) 41](#_Toc521515899)

[Figure 8: <Admin> Create Account for Partner (UC\_AD006) 42](#_Toc521515900)

[Figure 9: <Admin> Edit partner account (UC\_AD007) 44](#_Toc521515901)

[Figure 10: <Admin> Delete partner account (UC\_AD008) 46](#_Toc521515902)

[Figure 11: <Admin> View list film (UC\_AD009) 47](#_Toc521515903)

[Figure 12: <Admin> Add Film(UC\_AD010) 48](#_Toc521515904)

[Figure 13: <Admin> Edit Film (UC\_AD011) 51](#_Toc521515905)

[Figure 14 <Admin> Crawl film data (UC\_AD012) 53](#_Toc521515906)

[Figure 15: <User> Overview Use Case 54](#_Toc521515907)

[Figure 16: <Partner> View all cinema manager account (UC\_P002) 55](#_Toc521515908)

[Figure 17: <Partner> Add Cinema Manager Account (UC\_P003) 56](#_Toc521515909)

[Figure 18: <Partner>Delete Cinema Manager Account (UC\_P004) 58](#_Toc521515910)

[Figure 19: <Partner> Edit Cinema Manager Account (UC\_P005) 59](#_Toc521515911)

[Figure 20: <Partner> View all cinema (UC\_P006) 61](#_Toc521515912)

[Figure 21: <Partner> Arrange seat (UC\_P007) 62](#_Toc521515913)

[Figure 22: <Partner> Add Room for Cinema (UC\_P008) 64](#_Toc521515914)

[Figure 23: <Partner> Add New Cinema (UC\_P009) 66](#_Toc521515915)

[Figure 24: <Partner> Edit cinema (UC\_P010) 68](#_Toc521515916)

[Figure 25: <Partner> View Room List (UC\_P011) 70](#_Toc521515917)

[Figure 26: <Partner> View Room Detail (UC\_P012) 71](#_Toc521515918)

[Figure 27: <User> Overview Use Case 72](#_Toc521515919)

[Figure 28: <User> View Profile (UC\_U002) 73](#_Toc521515920)

[Figure 29: <User> View Booking History (UC\_U004) 74](#_Toc521515921)

[Figure 30: <Customer> Overview Use Case 75](#_Toc521515922)

[Figure 31: <Customer> Register (UC\_C001) 76](#_Toc521515923)

[Figure 32: <Customer> Filter Ticket (UC\_C002) 78](#_Toc521515924)

[Figure 33: <Customer>Choose type of seat and quality (UC\_C003) 80](#_Toc521515925)

[Figure 34: <Customer> Choose Seat (UC\_C004) 81](#_Toc521515926)

[Figure 35: <Customer> Checkout (UC\_C005) 83](#_Toc521515927)

[Figure 36: <Customer> Resell ticket (UC\_C006) 85](#_Toc521515928)

[Figure 37: <Customer> Change ticket (UC\_C007) 87](#_Toc521515929)

[Figure 38: <Cinema Manager> Overview Use Case 89](#_Toc521515930)

[Figure 39: <Cinema manager > Basic add movie schedule (UC\_M002) 89](#_Toc521515931)

[Figure 40: <Cinema manager > Advantage schedule (UC\_M003) 91](#_Toc521515932)

[Figure 41: <Cinema manager > View dashboard (UC\_M004) 93](#_Toc521515933)

[Figure 42: Initial <System> Overview Use Case 94](#_Toc521515934)

[Figure 43: <System> Notification about schedule (UC\_S002) 95](#_Toc521515935)

[Figure 44: <System> suggest film for user (UC\_S003) 96](#_Toc521515936)

[Figure 45: Conceptual Diagram 99](#_Toc521515937)

[Figure 46: Web Application Architecture 101](#_Toc521515938)

[*Figure 47: Web Service Application Architecture* 102](#_Toc521515939)

[*Figure 48: Mobile Application Architecture* 103](#_Toc521515940)

[Figure 49: Component Diagram 104](#_Toc521515941)

[Figure 50: Class Diagram 105](#_Toc521515942)

[Figure 51: Sequence Diagram - <User> Booking ticket 106](#_Toc521515943)

[Figure 52: Sequence Diagram - <User> Resell ticket 107](#_Toc521515944)

[Figure 53: Sequence Diagram - <User> Change ticket 107](#_Toc521515945)

[Figure 54: Sequence Diagram - <User> User register 107](#_Toc521515946)

[Figure 55: Sequence Diagram - <User> User login 107](#_Toc521515947)

[Figure 56: Sequence Diagram - <User> User logout 108](#_Toc521515948)

[Figure 57: Sequence Diagram - <User> Edit profile 108](#_Toc521515949)

[Figure 58: Sequence Diagram - <User> View booking ticket history 108](#_Toc521515950)

[Figure 59: Sequence Diagram - <Partner> Partner Login 109](#_Toc521515951)

[Figure 60: Sequence Diagram - <Partner> Partner create new cinema 109](#_Toc521515952)

[Figure 61: Sequence Diagram - <Partner> Partner add room for cinema 109](#_Toc521515953)

[Figure 62: Sequence Diagram - <Partner> Partner edit cinema information 109](#_Toc521515954)

[Figure 63: Sequence Diagram - <Partner> Partner add cinema manager account 110](#_Toc521515955)

[Figure 64: Sequence Diagram - <Partner> Partner edit cinema manager account information 110](#_Toc521515956)

[Figure 65: Sequence Diagram - <Partner> Partner delete cinema manager account 110](#_Toc521515957)

[Figure 66: Sequence Diagram - <Partner> Partner view report 110](#_Toc521515958)

[Figure 67: Sequence Diagram - <Admin> Admin login 111](#_Toc521515959)

[Figure 68: Sequence Diagram - <Admin> Admin add a film 111](#_Toc521515960)

[Figure 69: Sequence Diagram - <Admin> Admin edit film information 111](#_Toc521515961)

[Figure 70: Sequence Diagram - <Admin> Admin add group cinema 111](#_Toc521515962)

[Figure 71: Sequence Diagram - <Admin> Admin edit group cinema information 112](#_Toc521515963)

[Figure 72: Sequence Diagram - <Admin> Admin add partner account 112](#_Toc521515964)

[Figure 73: Sequence Diagram - <Admin> Admin edit partner account information 112](#_Toc521515965)

[Figure 74: Sequence Diagram - <Admin> Admin delete partner account 112](#_Toc521515966)

[Figure 75: Activity Diagram - Login 113](#_Toc521515967)

[Figure 76: Activity Diagram – Booking Ticket Web Application 114](#_Toc521515968)

[Figure 77: Activity Diagram – Booking Ticket Mobile Application 115](#_Toc521515969)

[*Figure 78: Entity Relationship Diagram (ERD)* 133](#_Toc521515970)

[Figure 79: Algorithms - Select group of seat on room's matrix - *a* 140](#_Toc521515971)

[Figure 80: Algorithms - Select group of seat on room's matrix - *b* 141](#_Toc521515972)

[Figure 81: Algorithms - Select group of seat on room's matrix - *c* 141](#_Toc521515973)

[Figure 82: Algorithms - Select group of seat on room's matrix - *d* 142](#_Toc521515974)

[Figure 83: Algorithms - Select group of seat on room's matrix - *e* 142](#_Toc521515975)

[Figure 84: Physical Diagram 143](#_Toc521515976)

[*Figure 85: Backend Architecture Implementation* 147](#_Toc521515977)

[*Figure 86: Mobile Application Architecture Implementation* 149](#_Toc521515978)

[*Figure 87: Web Architecture Implementation* 150](#_Toc521515979)

[Figure 88: Communication Diagram – User purchase ticket 151](#_Toc521515980)

[Figure 89: Communication Diagram – Admin- Admin Manage Film 152](#_Toc521515981)

[Figure 90: Communication Diagram – Admin- Admin Manage Partner 152](#_Toc521515982)

[Figure 91: Communication Diagram – Admin- Admin Manage Group of Cinema 152](#_Toc521515983)

[Figure 92: Communication Diagram – Partner - Partner Manage Cinema 153](#_Toc521515984)

[Figure 93: Communication Diagram – Partner - Partner Add Room 153](#_Toc521515985)

[Figure 94: Communication Diagram – Cinema Manager – Cinema Manager Manage Schedule 153](#_Toc521515986)

# Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| OCBS | Online Cinema Booking System |
| UI | User Interface |
| UX | User Experience |
| MVVM | Model – View – View Model |
| ORM | Object Relation Mapping |
| TDD | Test Driven Development |

1. Introduction
   1. Project Overview

* Official name: Online cinema booking system
* Vietnamese name: Hệ thống đặt vé xem phim trực tuyến
* Abbreviation: OCBS
* Product Type: Website and Mobile Applications
* Start Date: 21/05/2018
* End Date: 31/08/2018
  1. Introduction

The demand for seeing movie at cinema is growing fast, along with the rapid rising of smartphones and websites. Our team decided to develop a system that would allow users to book movie tickets easily from all our partners. The system also helps partners manage better their business by our revenue report’s system.  We allow the customer to control their business on our system. Besides that, we also to make our users more satisfied with us by trying to improve old function. We also add some new features for example: we allowing customer to resell their ticket to other customer in case they are busy and want to resell it.

* 1. Current Situation

In ordinary life, when you want to watch movies in the cinema, you have to buy tickets but you do not have time. Remember the time when you had to stand in a long queue outside the theatre to book movie tickets.

More and more cinema system was born and doesn’t has the solution for selling ticket online and also the system to manage their business.

There was very few online booking ticket page that collect data from many cinema system to combine in only one single page.

How do you not go to the cinema but have the ticket and have the best seats?

Nowadays, the force is strong to jump online or use highly accessible mobile apps to book tickets to the latest trending film. However, many cinemas in the market of Vietnam do not have general system for booking tickets. Customers must buy ticket offline or find another many website of Film Company to buy it. Therefore, they still look for a solution to enable moviegoers to purchase tickets online.

Nobody can deny that today’s society is one of convenience. With the technology sector continually looking to humor our desire to save time and increase efficiency we’ve become used to having access to the world through our fingertips. Nevertheless, many websites do not enough demand of customer. We’ve all heard about the highly disgruntled moviegoers about some problem for this buy the tickets online.

* 1. Problem Definition

Many entertainment companies want to enter to the cinema field and do not have cinema management software and solution for selling ticket online.

It's difficult for user to go to a lot of web page of each cinema to looking for online suitable ticket.

Many systems now doesn't allow customer to resell ticket when they are busy.

The current system does not allow changing ticket in a time interval (example: Before 4 hours of movie screening, users can change tickets to the later show time in day).

* 1. Proposed Solution

To the problems that existing systems have, we are developing a system that helps customers who are having trouble in finding a compatible service in booking cinema tickets. Customers who use our system will have more choices in how they can handle their purchased tickets.

We provide full package solution for our partner to manage their group cinema and each cinema in the group by their own. From add a new cinema, add schedule, view report to the booking ticket online system for real online customer. Our online booking ticket page for customer that integrates all the information of every partner that connects with us to compile into one single page. The other partner already has the management and online system, can also connect with us for the new marketing strategy when their information was public with other partner on our single page.

Cinemas which use our service will have less empty seats due to how our system works.

* + 1. Featured functions
* Web booking ticket online application:
  + Provides the ability to resell or change the tickets for customers who cannot come on time.
  + Customer can easily booking ticket online without logged in.
  + Cinemas auto suggest best film based on evaluation system.
  + Customer can resell ticket online easily.
* Mobile application:
  + QR code will be provided for each ticket purchased and checked at the movie room gate
  + Customer can resell ticket and change show time for already bought ticket.
* Web management application (for cinema manager):
  + Cinemas give the cinema manager the report on the dashboard each time they logged in to the management system.
  + System can automatically suggest schedule for cinema manager. With our suggestion to ensure bring greatest benefit to the partner.
* Web management application (for partner):
  + System allows partner to add new cinema and add room for cinema easily with beauty and convenient UI.
  + System allows partner to manage all cinema and employee belong with their group.
* Web admin application:
  + System has an ability to crawl film data from the internet.
  + Admin can add new group cinema belong to a partner who want to connect with us.
    1. Possible drawbacks
* The system is only all basic background function for big system. So they may be not perfect and need to be upgrade more and more.
* Customers who want to buy the tickets that are being buy by other customers have to contact the seller themselves to make the transaction.
  1. Functional Requirements

The functional requirements are listed as below:

* Customer component:
  + Booking ticket: customers can choose the ticket base on film, cinema and time; and check out the ticket.
  + Resell ticket: customers can resell ticket for the other if they are busy.
  + Change time: customers can change time for a ticket that they already bought.
  + Register: customers can register new account to become a user
* User component:
  + User can do whatever customer can.
  + Login: User can login to the system.
  + Manage profile: User can view and edit their profile, view booking history.
* Partner component:
  + Partner Login: partner can login to manage their cinema group
  + Add Cinema: partner can add new cinema
  + Add Room: partner can add room for cinema that they have created.
  + Arrange Seat: partner can arrange seat for the room in cinema
  + View statistic and report: partner can see the user statistic and revenue report.
* Cinema manager component:
  + Add Schedule for the cinema
  + View weekly ticket report on their dashboard.
  + Automatically suggest for the optimize schedule that bring the greatest profit for the cinema.
* Admin component:
  + Admin login: Admin can login to manage partner and the system.
  + Add partner: Admin can create new group of cinema of new partner
  + Add Film: Admin can add new film for partner to create schedule
  + Manage Partner Account: Admin can add, disable, and change information of partner’s account.
* System component:
  + Notify: the system will notify to user before their ticket coming to show 1 hour.
  + Automatically suggest best-selling film for user.
  1. Roles and Responsibilities

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Full Name** | **Role** | **Position** | **Contact** |
| 1 | Nguyễn Thị Cẩm Hương | Project Manager | Supervisor | huongntc2@fpt.edu.vn |
| 2 | Lê Công Danh | Developer | Leader | danhlcse61769@fpt.edu.vn |
| 3 | Lưu Lạc | Developer | Member | laclse62056@fpt.edu.vn |
| 4 | Hồ Sĩ Hoàng | Developer | Member | hoanghsse61397@fpt.edu.vn |
| Table 1: Roles and Responsibilities | | | | |

1. Software Project Management Plan
   1. Problem Definition
      1. Name of this Capstone Project

* Official name: Online cinema booking system
* Vietnamese name: Hệ thống đặt vé xem phim trực tuyến
* Abbreviation: OCBS
  + 1. Problem Abstract
* The demand for watching movies at the cinema is rising. Customer want a fast and convenient way to buy tickets.
* More and more cinema system was born and doesn’t has the solution for selling ticket online and also the system to manage their business.
* There was very few online booking ticket page that collect data from many cinema system to combine in only one single page.
* We offer a convenient online booking movie ticket application for both web and mobile.  We always want the best service for our users. So, we have added some new features to help those buying tickets easier. For example, we allow them to pay or resell tickets in case they are busy.
* We also care a lot about our partner. We provide more support to our partners by offering statistical reports that help them better manage their cinemas.
  + 1. Project Overview
       1. Current diagnostic diseases

Below are the problems encountered in this project:

* **Lack of UI, UX knowledge:** Our team members all study Information System and no one have good understanding about UI and UX of a standard E-Commerce application. That is the reason why spending time to study popular websites UI, UX is quite important.
* **Many similar systems on the market:** There are many systems like us in the market. The challenge is we have to do better than what they already have. Besides that, we need to analyses their system’s weaknesses and add new feature to overcome it.
* **We do not know a lot about our future partner:** We need to find out what our partner need and provide it. We analyses information collect from the system to support for our partner. The biggest purpose for partner is to increase their revenue.
  + - 1. The Proposed System

Our system includes four main subsystems: one online websites for customer and user to booking ticket, one website for partner and cinema manager to manage their movie theater, another website for admin to manage the system. For convenient, we also have mobile application system for user and customer to booking ticket.

* + - * 1. Web Application

Website is a common system for user (customers), partner, cinema manager and admin. Website provide following features:

* For Customer:
  + Customer can booking ticket online: they can choose which film, cinema, time, seat that they like and then checkout the get the ticket.
  + Customer can resell ticket when they are busy or got in to urgent situation.
* For User:
  + User can login to the system.
  + User can manage their profile: user can view and edit profile.
  + User can see their booking history.
  + User can also have same ability of customer.
* For Cinema Manager:
  + Cinema Manager can login to the system.
  + Cinema Manager can view their cinema report on dashboard.
  + Cinema Manager can add schedule for their cinema.
* For Partner:
  + Partner can do what cinema Manager can do.
  + Partner can manage the cinema in their system.
  + Partner can add new cinema: they can add new cinema and then add room and arrange seat to this room.
* For Admin:
  + Admin can login to admin’s system.
  + Admin can add and manage film list.
  + Admin can add new group cinema belong to partner.
  + Admin can manage partner account: They can add new partner and add new partner type of seat. They also can disable change information of partner’s account.
    - * 1. Mobile application

The mobile application is mostly for users and customer to:

* + System will automatic filter film follow the action that you interactive with the system.
  + Manage profile only for user.
  + Resell and change time for the ticket.
    - 1. Boundaries of the System
* Everybody who interacts with the system has to equip enough device as follow:
  + Computer with internet connection.
  + An android smartphone with android 6.0 or above.
* The language of this system is both Vietnamese and English.
* The complete product includes:
  + Website application for customer, user, partner, cinema manager and admin
  + Mobile application for user.
* Every Cinema partner can corporate with us to manage their cinema better and improve their marketing.
  + - 1. Future plans
* System use machine learning to find out which film the user would like to see and then recommend it to them.
* We will integrate with GPS finding cinema near user on mobile application.
* We will integrate with more platforms of payment.
* We will integrate a forum about movie on this website.
  + - 1. Development Environment
         1. Hardware requirement
* **For server:**

|  |  |  |
| --- | --- | --- |
| **Hardware** | **Minimum Requirements** | **Recommended** |
| **Internet Connection** | Cable (4 Mbps) | Cable (8 Mbps) |
| **Operating System** | Windows Vista, 7, 8 | Windows 7, 8 |
| **Computer Processor** | Intel® Core i3 1.4GHz | Intel® Core(TM) i5 CPU , M 460 @ 2.53GHz |
| *Table 2: Hardware requirement for Server* | | |

* **For web:**

|  |  |  |
| --- | --- | --- |
| **Hardware** | **Minimum Requirements** | **Recommended** |
| **Internet Connection** | **512Kbps** | **8 Mbps** |
| **Operating System** | Windows Vista, 7, 8 | Windows 7, 8 |
| **Computer Processor** | 1 GHz | Intel® Core(TM) i5 CPU , M 460 @ 2.53GHz |
| **Computer Memory** | 1GB of RAM | 3GB of RAM or more |
| *Table 3: Hardware requirement for Web* | | |

* **For mobile:**

|  |  |  |
| --- | --- | --- |
| **Hardware** | **Minimum Requirements** | **Recommended** |
| **Internet Connection** | 512Kbps | Wi-Fi Connection 12MB |
| **Operating System** | Android 6.0 | Android 6.0 |
| **Hardware** | NFC supported | NFC supported |
| **Memory** | 128MB of RAM | 1GB of RAM or more |
| *Table 4: Hardware requirement for Mobile* | | |

* + - * 1. Software requirement

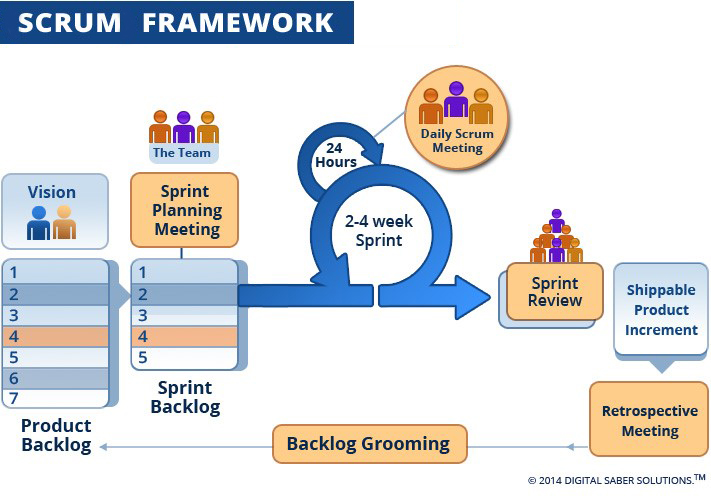
|  |  |
| --- | --- |
| **Software** | **Name / Version** |
| Operating system | Windows 7 or above |
| Environment | NET Framework |
| Modeling tool | Microsoft Visual 2013 |
| IDE | Visual studio 2013, Android studio 3.1.2 |
| DBMS | Microsoft Sql Server 2012 |
| Source control | Github Desktop |
| Web browser | Chrome 42 or above |
| *Table 5: Software requirement* | |

*Table 5: Software requirement*

* 1. Project organization
     1. Software Project Model

This project is developed for using Scrum model – part of an agile framework for Software development project. Our team choose Scrum model because of these following reasons:

* We had constantly changing requirements, as soon as we got something on paper, the stakeholder would make a change.
* Prototypes are delivered frequently for evaluation, usually weekly, rather than months.
* Take fewer risks when there is a change in requirement.
* All members must work together in order to avoid misunderstanding or miscommunication.
* Able to study new skills or knowledge at the same time as developing.



*Figure 1: Scrum Process*

(<http://www.digitalsaber.com/process/agile-scrum-framework/>)

* + 1. Roles and Responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Full Name** | **Role in Group** | **Responsibilities** |
| 1 | Nguyễn Thị Cẩm Hương | Product owner | * Specify scope and user requirement. * Supervise the development progress. * Provide professional techniques and business analysis support. |
| 2 | Lê Công Danh | Scrum master | * Managing process * Designing database * Clarifying requirements * Prepare documents * GUI design * Create test plan * Coding * Testing |
| 3 | Lưu Lạc | Team member | * Designing database * Clarifying requirements * Prepare documents * GUI design * Create test plan * Coding * Testing |
| 4 | Hồ Sĩ Hoàng | Team member | * Designing database * Clarifying requirements * Prepare documents * GUI design * Create test plan * Coding * Testing |
| *Table 6: Roles and Responsibilities* | | | |

* + 1. Tools and Techniques

|  |  |  |
| --- | --- | --- |
|  | **Tools** | **Techniques** |
| Front-end | Bracket | * HTML5 * CSS3 * ES6 * AngualarJS * Jquery |
| Back-end | Visual studio 2013 | * AJAX * JSON * ASP.NET MVC |
| Web server |  | * N/A |
| Mobile Application | Android studio | * Java 8 * Android SDK |
| Database management system | SQL SERVER 2012 | N/A |
| *Table 7: Tools and Techniques* | | |

* 1. Project Management Plan
     1. Product Backlog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Story ID** | **Features** | **Task ID** | **Task description** | **Sprint** |
| 1 | Create Product Backlog | 1 | Create Product Backlog | 1 |
| 2 | Write Introduction document | 2 | Write Introduction document | 1 |
| 3 | Write project management plan | 3.1 | Problem definition | 1 |
| 3.2 | Project organization | 1 |
| 3.3 | Project management plan | 1 |
| 3.4 | Coding convention | 1 |
| 3.5 | Review document | 1 |
| 4 | Building system structure | 4.1 | Building back-end structure | 1 |
| 5 | Write software requirements | 5.1 | User requirement specification | 2 |
| 5.2 | Use case diagram | 2 |
| 6 | Write software design and documentation | 6.1 | System architectural design | 2 |
| 6.2 | Component diagram | 2 |
| 6.3 | Class diagram | 2 |
| 6.4 | Sequence diagram | 2 |
| 6.5 | Database design | 2 |
| 6.6 | Entity relationship diagram | 2 |
| 6.7 | Design user interface | 2 |
| 6.8 | Test plan and test case | 2 |
| 7 | Implementation | 7.1 | User registration | 3 |
| 7.2 | User login | 3 |
| 7.3 | User logout | 3 |
| 7.4 | User change them information | 3 |
| 7.5 | Booking ticket | 3 |
| 7.6 | Check out | 3 |
| 7.7 | Resell ticket | 3 |
| 7.8 | Return ticket | 3 |
| 7.9 | User cancel ticket | 3 |
| 7.10 | Displays a list of cinemas | 3 |
| 7.11 | Display a news page | 3 |
| 7.12 | Filter movie by title, theater, date, show time | 3 |
| 7.13 | Display movie (now show, coming soon) | 3 |
| 7.14 | Displays a movie's schedule for a specific movie | 3 |
| 7.15 | Displays theater’s information for a specific theater | 4 |
| 7.16 | Displays a list of movie schedule in a specific theater | 4 |
| 7.17 | Partner login | 4 |
| 7.18 | Partner create cinema | 4 |
| 7.19 | Partner add room to cinema | 4 |
| 7.20 | Partner arrange seat in room | 4 |
| 7.21 | Partner add film schedule | 4 |
| 7.22 | Partner view report | 4 |
| 7.23 | Partner view user statistic | 4 |
| 7.24 | Admin login | 4 |
| 7.25 | Amin add partner | 4 |
| 7.26 | Admin add partner type of seat | 4 |
| 7.27 | Admin manage the list of cinemas | 4 |
| 8 | Software testing | 8.1 | Unit test | 5 |
| 8.2 | Integration test | 5 |
| 8.3 | System test | 5 |
| 9 | Quality assurance | 9.1 | Quality assurance for back-end | 5 |
| 9.2 | Quality assurance for web | 5 |
| 9.3 | Quality assurance for mobile | 5 |
| 10 | Software user’s manual | 10.1 | Installation guide | 6 |
| 10.2 | User’s guide | 6 |
| Table 8: Product Backlog | | | | |

* + 1. Sprint Backlog
       1. Sprint 1 (14/5/2018 -27/5/2018) Project Initiation
          1. Goal
* Sprint 1 must complete following tasks:
* Create Product Backlog
* Create Introduction document:
* Write Introduction document
* Review Introduction document
* Create project management plan:
* Problem definition
* Project organization
* Project management plan
* Coding convention
  + - Review document
* Build system structure:
* Backend Structure
* Web Structure
* Mobile Structure
  + - * 1. Development

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Task** | **Implement** | **Reviewer** |
| 1 | Create Product Backlog | LacL | DanhLC |
| 2 | Write Introduction document | HoangHS | DanhLC |
| 3 | Problem definition | DanhLC | LacL |
| 4 | Project organization | HoangHS | DanhLC |
| 5 | Project management plan | DanhLC | LacL |
| 6 | Coding convention | HoangHS | LacL |
| 7 | Review document | DanhLC | HuongNTC |
| 8 | Backend Structure | DanhLC, HoangHS | LacL |
| 9 | Web Structure | DanhLC | HuongNTC |
| 10 | Mobile Structure | LacL | HuongNTC |
| 11 | Mobile prototype | LacL | HuongNTC |
| *Table 9: Sprint 1 Development* | | | |

* + - 1. Sprint 2 (28/5/2018 - 3/6/2018) Software requirement and basic features
         1. Goal
* Sprint 2 must complete the following tasks:
* Write software requirements:
* User Requirement Specification
* Use case diagram
* Write software design and documentation:
* System architectural design
* Component diagram
* Class diagram
* Sequence diagram
* Database design
* Entity relationship diagram
* Design user interface
* Test plan and test case
  + - * 1. Development

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | User Requirement Specification | DanhLc, LacL | HuongNTC |
| 3 | Use case diagram | DanhLC | HuongNTC |
| 5 | System architectural design | DanhLC, LacL | HuongNTC |
| 6 | Component diagram | DanhLC | HuongNTC |
| 7 | Class diagram | LacL, HoangHS | HuongNTC |
| 8 | Sequence diagram | LacL, HoangHS | HuongNTC |
| 9 | Database design | DanhLC | HuongNTC |
| 10 | Entity relationship diagram | LacL | HuongNTC |
| 11 | Design user interface | DanhLC(web), LacL(mobile) | HuongNTC |
| 12 | Test plan and test case | LacL | HuongNTC |
| 1 | User Requirement Specification | DanhLc, LacL | HuongNTC |
| 3 | Use case diagram | DanhLC | HuongNTC |
| 5 | System architectural design | DanhLC, LacL | HuongNTC |
| *Table 10: Sprint 2 Development* | | | |

* + - 1. Sprint 3 (4/6/2018 - 17/6/2018) Software Design Description and main features
         1. Goal
* Sprint 3 must complete the following tasks:
* Implementation:
* User registration
* User login
* User logout
* User change them information
* Booking ticket
* Check out
* Resell ticket
* Return ticket
* User cancel ticket
* Displays a list of cinemas
* Display a news page
* Search movie by title, theater, date, showtime
* Display movie (now show, coming soon)
  + - * Displays a movie's schedule for a specific movie
        1. Development

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Task** | **Implement** | **Reviewer** |
| 1 | User registration | HoangHS | LacL |
| 2 | User login | HoangHS | LacL |
| 3 | User logout | HoangHS | DanhLC |
| 4 | User change them information | HoangHS | LacL |
| 5 | Booking ticket | DanhLC | LacL |
| 6 | Check out | DanhLC | LacL |
| 7 | Resell ticket | DanhLC | LacL |
| 8 | Return ticket | DanhLC | LacL |
| 9 | User cancel ticket | DanhLC | LacL |
| 10 | Displays a list of cinemas | LacL | DanhLC |
| 11 | Display a news page | LacL | DanhLC |
| 12 | Search movie by title, theater, date, show time | LacL | DanhLC |
| 13 | Display movie (now show, coming soon) | LacL | DanhLC |
| 14 | Displays a movie's schedule for a specific movie | LacL | DanhLC |
| *Table 11: Sprint 3 Development* | | | |

* + - 1. Sprint 4 (18/6/2018 – 8/7/2018) Finish up coding
         1. Goal
* Sprint 4 must complete following tasks:
* Implementation:
* Displays theater’s information for a specific theater
* Displays a list of movie schedule in a specific theater
* Partner login
* Partner create cinema
* Partner add room to cinema
* Partner arrange seat in room
* Partner add film schedule
* Partner view report
* Partner view user statistic
* Admin login
* Amin add partner
* Admin add partner type of seat
* Admin manage the list of cinemas
* Admin manage the list of movies
  + - * 1. Development

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Task** | **Implement** | **Reviewer** |
| 1 | Displays theater’s information for a specific theater | DanhLC | LacL |
| 2 | Displays a list of movie schedule in a specific theater | DanhLC | LacL |
| 3 | Partner login | HoangHS | LacL |
| 4 | Partner create cinema | HoangHS | LacL |
| 5 | Partner add room to cinema | DanhLC | LacL |
| 6 | Partner arrange seat in room | DanhLC | LacL |
| 7 | Partner add film schedule | LacL | DanhLC |
| 9 | Partner view report | DanhLC | LacL |
| 10 | Partner view user statistic | HoangHS | LacL |
| 11 | Admin login | HoangHS | DanhLC |
| 12 | Amin add partner | LacL | DanhLC |
| 13 | Admin add partner type of seat | LacL | DanhLC |
| 14 | Admin manage the list of cinemas | LacL | DanhLC |
| 15 | Admin manage the list of movies | LacL | DanhLC |
| *Table 12: Sprint 4 Development* | | | |

* + - 1. Sprint 5 (8/7/2018 – 22/7/2018) Software Test Document and Quality Assurance
         1. Goal
* Sprint 5 must complete the following tasks:
* Software testing:
* Unit test
* Integration test
* System test
* Quality assurance:
* Quality Assurance for Backend
* Quality Assurance for Web
* Quality Assurance for Mobile
  + - * 1. Development

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Task** | **Implement** | **Reviewer** |
| 1 | Unit test | All | Self review |
| 2 | Integration test | LacL | DanhLc |
| 3 | System test | DanhLC | LacL |
| 4 | Quality Assurance for Backend | HoangHS | DanhLC |
| 5 | Quality Assurance for Web | HoangHS | DanhLC |
| 6 | Quality Assurance for Mobile | LacL | DanhLC |
| *Table 13: Sprint 5 Development* | | | |

* + - 1. Sprint 6 (22/7/2018 – 5/8/2018) Software User’s Manual
         1. Goal
* Sprint 6 must complete the following tasks:
* Installation Guide
* User’s Guide
  + - * 1. Development

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Task** | **Implement** | **Reviewer** |
| 1 | Installation Guide | LacL, HoangHS | DanhLC |
| 2 | User’s Guide | DanhLC | LacL |
| *Table 14: Sprint 6 Development* | | | |

* + 1. Deliverables

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Deliverable** | **Deliverable date** | **Deliverable location** | **Note** |
| 1 | Introduction Document, Task list |  | FU - LMS | Report No. 1 |
| 2 | Software Project Management Plan |  | FU – LMS | Report No. 2 |
| 3 | Software Requirements Specification |  | FU – LMS | Report No. 3 |
| 4 | Software Design Description |  | FU – LMS | Report No. 4 |
| 5 | Software Test Documentation Guide Implementation (Coding) |  | FU – LMS | Report No. 5 |
| 6 | Software User’s Manual |  | FU - LMS | Report No. 6 |
| *Table 15: Deliverables* | | | | |

* For each Sprint, deliverables are potentially shippable products, which can be a part of document or prototype implemented based on the project’s core flow without any constraints.
* Each Sprint has a fixed duration of two weeks.
  + 1. All Meeting Minutes

All meeting documents could be found [here](https://drive.google.com/drive/folders/0B9QIp5ehkI6rXzVtdE9OMVNtTjg)

* 1. Coding Convention

C#.NET : Using for backend website:

* Naming convention:
* In short examples that do not include [using directives](https://github.com/DanhlcGitHub/CinemaTicket), use namespace qualifications. If you know that a namespace is imported by default in a project, you do not have to fully qualify the names from that namespace. Qualified names can be broken after a dot (.) if they are too long for a single line.
* You do not have to change the names of objects that were created by using the Visual Studio designer tools to make them fit other guidelines.
* Indentation:
* These formatting rules concern the use of indentation to format code.
* The following table shows the rule names, applicable languages, default values, and first supported version of Visual Studio.

JavaScript (ES6): Using for frontend website

* Naming convention:
* Use camel-case style for variables and functions name (Lowercase for first letter)
* Use camel-case for classes (Uppercase for first letter)
* Indentation:
* Use two spaces per indentation level
* Do not use “;” for separate statements and expressions (ES6)
* Avoid single-line method
* Classes:
* Classes should be declared using class keyword for ES6. Using function to declares classes in ES6 is bad
* Use constructor function for initialize a class
* Use export and import to reuse modules

-    Use **this** to refer current class instance

Java (Android): Using for mobile development

* Naming convention:
* Use **camel-case** style for variables and functions name (Lowercase for first letter)
* Use **camel-case** for classes (Uppercase for first letter)
* Indentation:
* Use four **spaces** per indentation level
* User eight spaces to continue indent
* Use “;” for separate statements and expressions
* Classes:
* Classes should be declared using class keyword
* Using public, private, protected to declare protection level of instance variable and function
* -    Use **this** to refer current class instance

References:

* [1] C#.NET: https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/inside-a-program/coding-conventions
* [2] ES6: <https://github.com/elierotenberg/coding-styles/blob/master/es6.md>
* [3] Java: <http://www.oracle.com/technetwork/java/codeconvtoc-136057.html>

1. Software Requirement Specifications
   1. User Requirement Specification
      1. Admin Requirement

Admin is the person who manages all the films and partners in the system. Admin can do the following functions:

* Manage films:
* Add new films
* Edit films
* View list films
* Manage Partner:
* Add new partner
* Edit partner information
* Disable partner account
* Manager Group Cinema:
* Add new group cinema
* Edit group cinema information
* Crawl film data:
* Crawl film by month.
* Review and add film to real system.
  + 1. Partner Requirement

Partner is the person who manages a group of cinemas that belong to a brand. Partner can do the following functions:

* Manage cinema
* Add cinema manager account.
* Add room and arrange seat for this room.
* Edit cinema manager account.
* Manage cinema account:
* Add cinema manager account.
* Edit cinema manager account
  + 1. Cinema Manager Requirement

Cinema manager is the person who manages 1 cinema of the cinemas that the partner own. Cinema manager can do the following functions:

* Add movie schedule
* View Report
  + 1. User Requirement

Users are the people who use the service that have already registered. User can do the following functions:

* Manage user profile:
* View profile
* View booking history
  + 1. Customer Requirement

Customers are the people who use the service that may or may not have registered. Customer can do the following functions:

* Register
* Book ticket
  1. Software Requirement Specification
     1. External Interface Requirement
        1. User Interface
* The user interface uses Vietnamese as main language for users and English for Staff, Manager and Admin on both Mobile and Web application.
* The user interface displays best on 1024x768 and above screen size
  + - 1. Hardware Interface
* Android Smartphone: Nexus 4 or later.
  + - 1. Software Interface
* Web application: work with Firefox (v30 or above), Chromes (v25 or above)
* Mobile Application: Android 4 (or later)
  + - 1. Communication Protocol
* Use HTTP protocol 1.1 for communication between the web browser and the web server.
* Use HTTP protocol 1.1 for communication between the mobile application and the web service.
* Use HTTP protocol 1.1 for communication between the server and the Microsoft service.
  + 1. System Overview Use Case

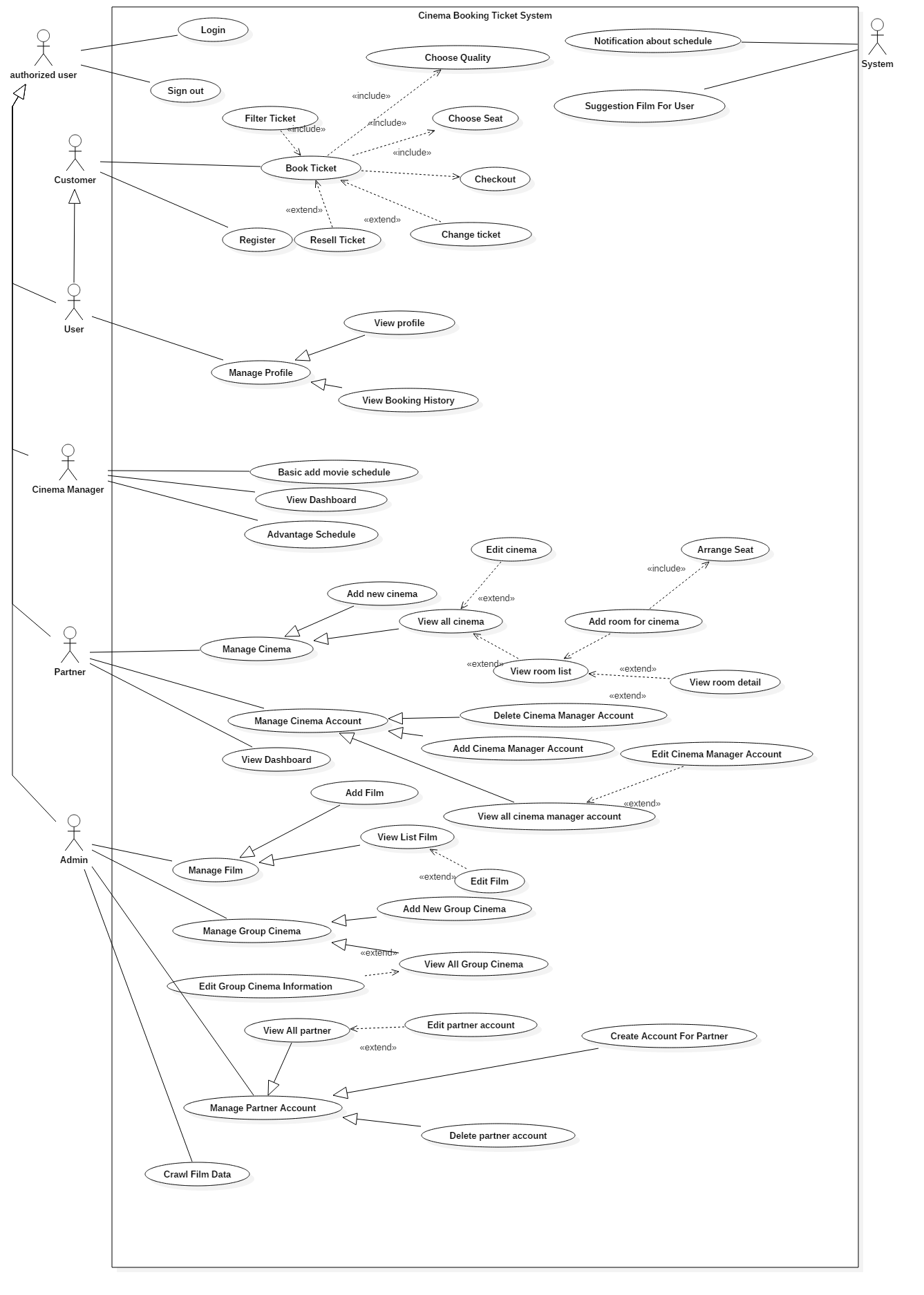


Figure 2: System Overview Use Case

* + 1. List of Use Case
       1. <Admin> Overview Use Case

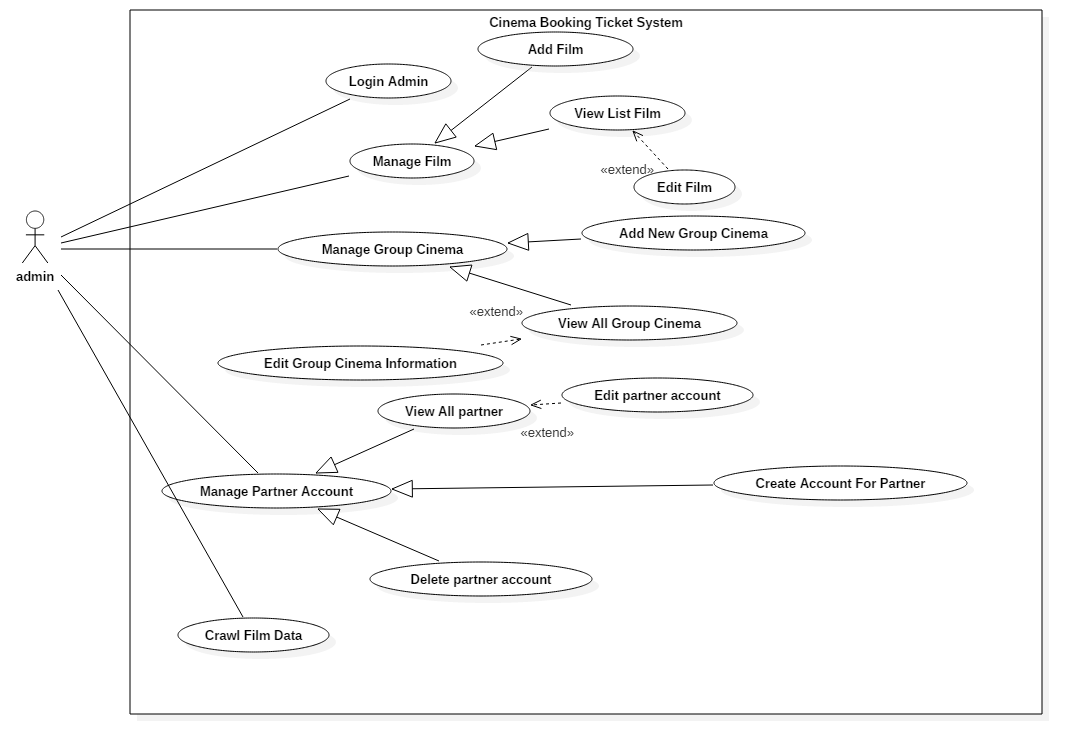


Figure 3: <Admin> Overview Use Case

* + - * 1. <Admin> View All Group Cinema (UC\_AD002)

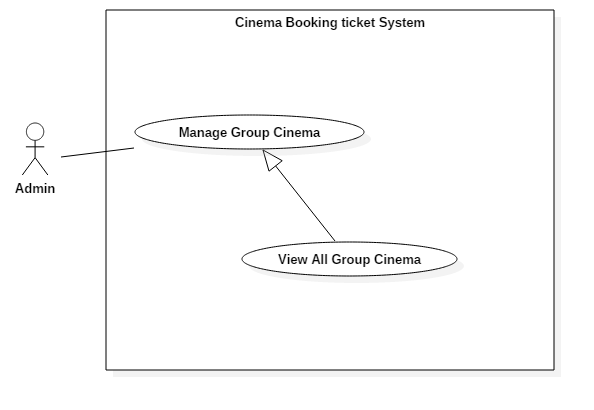


Figure 4: <Admin> View All Group Cinema (UC\_AD002)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_AD002** | | | |
| **Use Case No.** | UC\_AD002 | **Use Case Version** | 2.0 |
| **Use Case Name** | View all group cinema | | |
| **Author** | DanhLC | | |
| **Date** | 19/06/2018 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case allows actor to view all group cinema which already exist in the system. The list of all group cinemas will display in table format, each row of the table will display basic information of it.   **Goal:**   * Display all group cinema   **Triggers:**   * Actor sends command to view all group cinema   **Preconditions:**   * Actor must login as Admin role.   **Post Conditions:**   * **Success:** Action complete. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Actor click “Manage group cinema” button. | System will display list of all group cinema as table, each row of table will display following information:   * **name**: text, read only. * **phone**: text, read only. * **Email**: text, read only. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:** N/A | | | |
| Table 16: USE CASE - UC\_AD002 - <Admin> View all group cinema | | | |

* + - * 1. <Admin> Add new Group Cinema (UC\_AD003)

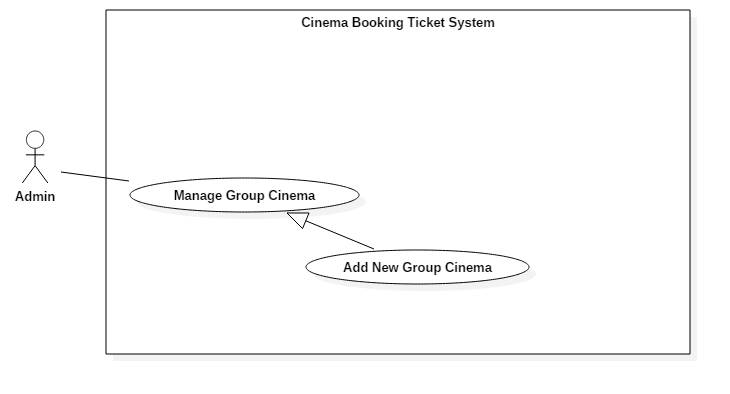


Figure 5: <Admin> Add new group cinema (UC\_AD003)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_AD003** | | | |
| **Use Case No.** | UC\_AD003 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add new group cinema | | |
| **Author** | DanhLC | | |
| **Date** | 19/06/2018 | **Priority** | High |
| **Actor:**   * Admin   **Summary:**   * This use case allows actor to add a new group cinema to the system. And when a cinema group was created, a Partner account will be created later to manage this group cinema.   **Goal:**   * This function helps actor create a new group cinema.   **Triggers:**   * Actor sends request to add new group cinema.   **Preconditions:**   * Actor must login admin first.   **Post Conditions:**   * **Success:** Action complete. * **Fail:** System will display message below.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User click “manage group cinema”. | System will display manage group cinema screen. | | 2 | User click “Add new group cinema”. | System requires information from admin:   * **Logo Picture**: hidden free text input, required, length 8–40 characters. * **Group cinema name:** free text input, required, length 8–40 characters. * **Phone**: free text input, required, length 9-11 digits. * **Address**: free text input, required, length 8-500 characters. * **Email**: free text input, required, length 8-40 characters. And must be in email format. | | 3 | Actor inputs information. |  | | 4 | Actor sends command to save.  [Alternative 1] | System validates information.  [Exception 1, 2, 3, 4, 5] | | 5 |  | System adds a new group cinema with inputted information and notify manager via email. |   **Alternative Scenario:**  *[Alternative 1]*   |  |  |  | | --- | --- | --- | | Step. | Actor Action | System Response | | 1 | Actor sends request to cancel. | System cancels adding new group cinema and redirect to home page. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Invalid Group cinema name format. | System shows error message: “group cinema name cannot be blank and must be 8-40 characters long”. | | 2 | Invalid Address format. | System shows error message: “Address cannot be blank and must be 8-40 characters long”. | | 3 | Invalid Email format | System shows error message: “Email cannot be blank and must be 8-40 characters long”. | | 4 | Invalid Phone format. | System shows error message: “Phone cannot be blank or contain alphabetical characters and must be 9-11 digits, ”. | | 5 | Invalid Logo Picture format | System shows error message: “Picture cannot be empty and must be in jpeg/jpg/png format”. |   **Relationships:**  **Business Rules:**   * Password is encrypted at server. * Partner account will be auto send back to partner via email. | | | |
| Table 17: USE CASE - UC\_AD003 - <Admin> Add new group cinema | | | |

* + - * 1. <Admin> Edit group cinema information account (UC\_AD004)

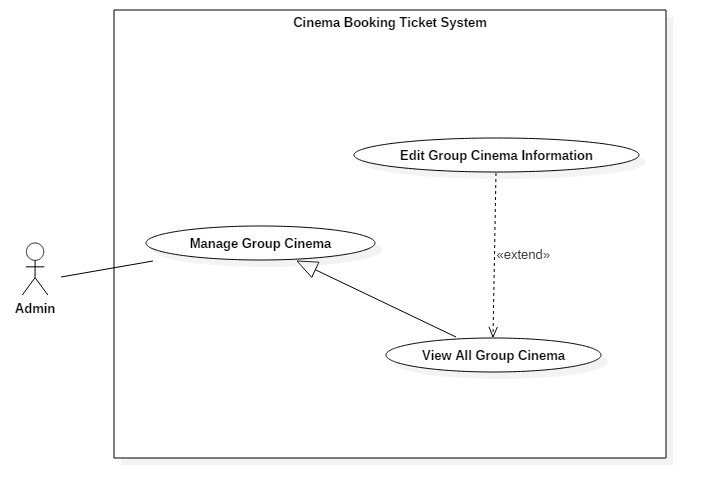


Figure 6: <Admin> Edit group cinema information(UC\_AD04)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_AD004** | | | |
| **Use Case No.** | UC\_AD004 | **Use Case Version** | 2.0 |
| **Use Case Name** | Edit group cinema information | | |
| **Author** | DanhLC | | |
| **Date** | 19/06/2018 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case helps actor to edit group cinema information, which already been existed in the system.   **Goal:**   * To change to information of this group cinema.   **Triggers:**   * Actor sends edit group cinema request.   **Preconditions:**   * User login into system as Admin. * Group cinema must already be existed.   **Post Conditions:**   * **Success:** Action complete. * **Fail:** System will display below messages   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home page, actor click “Manage group cinema” button | System show list group cinema in table format. | | 2 | Actor click “detail” button next to the group cinema want to edit. | System requires information from admin:   * **Logo Picture**: hidden free text input, required, length 8–255 characters. * **Group cinema name:** free text input, required, length 8–40 characters. * **Phone**: free text input, required, length 9-11 digits. * **Address**: free text input, required, length 8-500 characters. * **Email**: free text input, required, length 8-40 characters. And must be in email format. | | 3 | Actor click “save” button.  [Alternative 1] | System validates information.  [Exception 1, 2, 3, 4, 5] | | 4 |  | System edit selected group cinema with inputted information to database. |   **Alternative Scenario:**  [Alternative 1]   |  |  |  | | --- | --- | --- | | Step. | Actor Action | System Response | | 1 | Actor sends command to cancel. | System cancels edit group cinema action. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Invalid Group cinema name format. | System shows error message: “Store name cannot be blank and must be 8-40 characters long”. | | 2 | Invalid Address format. | System shows error message: “Address cannot be blank and must be 8-500 characters long”. | | 3 | Invalid Email format | System shows error message: “Email cannot be blank and must be 8-40 characters long”. | | 4 | Invalid Phone format. | System shows error message: “Phone cannot be blank or contain alphabetical characters and must be 9-11 digits”. | | 5 | Invalid Picture format | System shows error message: “Picture cannot be empty and must be in jpeg/jpg/png format”. |   **Relationships:** N/A  **Business Rules:** N/A | | | |
|  | | | |
| Table 18: USE CASE - UC\_AD004 - <Admin> Edit Group Cinema Information | | | |

* + - * 1. <Admin> View all Partner (UC\_AD05)

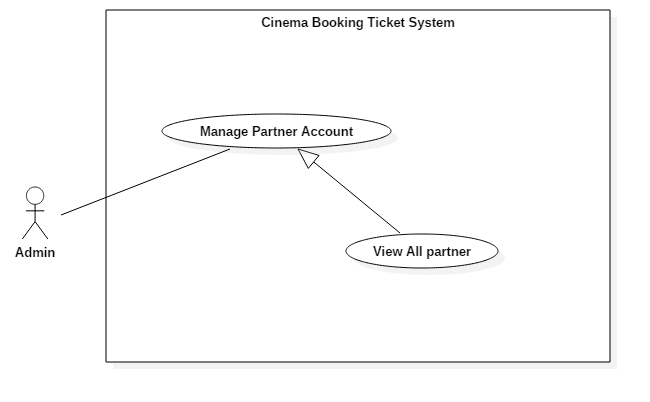


Figure 7: <Admin> View All Partner (UC\_AD005)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_AD005** | | | |
| **Use Case No.** | UC\_AD005 | **Use Case Version** | 2.0 |
| **Use Case Name** | View all Partner | | |
| **Author** | DanhLC | | |
| **Date** | 19/06/2018 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case allows actor to view all partner account that existing in the system. List of partners is displayed as table form and each row displays the basic information of a specific partner account.   **Goal:**   * Successfully actor can view all list of partner account.   **Triggers:**   * Actor sends command to view all partner account.   **Preconditions:**   * Actor must login as Admin role.   **Post Conditions:**   * **Success:** Action complete. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen actor click “manage partner account” | System displays a list of all partner account in a table form. Each row will display flowing information:   * **name**: text, read only. * **phone**: text, read only. * **Email**: text, read only. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:** N/A | | | |
| Table 19: USE CASE - UC\_AD005 - <Admin> View all Partner | | | |

* + - * 1. <Admin> Create Account for Partner (UC\_AD006)

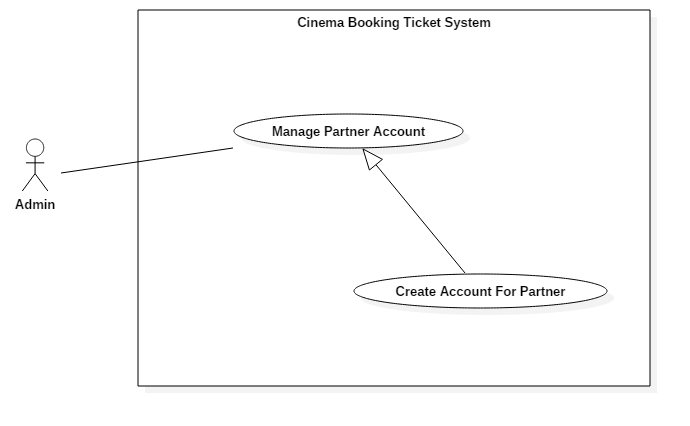


Figure 8: <Admin> Create Account for Partner (UC\_AD006)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_AD006** | | | |
| **Use Case No.** | UC\_AD006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Create account for partner | | |
| **Author** | DanhLC | | |
| **Date** | 19/06/2018 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case allows actor to create new account for partner. This account will associate with one and only one group cinema. The account has been created will be used to manage chosen group cinema.   **Goal:**   * Successfully added new Partner account will be added to the system.   **Triggers:**   * Actor sends request to add Partner account.   **Preconditions:**   * Actor must login as Admin role.   **Post Conditions:**   * **Success:** Action complete. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home page, actor click “Manage partner account” button. | System will display manage partner view. | | 2 | Actor click “add partner” button. | System will have required input following field:   * **Partner username:** free text input, required, length 8–40 characters. * **Phone**: free text input, required, length 9-11 digits. * **Email**: free text input, required, length 8-40 characters. * **Group cinema Id**: select a cinema from the dropdown list. | | 3 | Click button “add”.  [Alternative 1] | System validates information.  [Exception 1, 2, 3, 4, 5] | | 4 |  | System adds new partner account to database. |   **Alternative Scenario:**  *[Alternative 1]*   |  |  |  | | --- | --- | --- | | Step. | Actor Action | System Response | | 1 | Actor sends command to cancel. | System cancels adding new partner account. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Invalid Partner username format. | System shows error message: “Store name cannot be blank and must be 8-40 characters long”. | | 2 | Invalid Email format. | System shows error message: “Email cannot be blank and must be 8-40 characters long”. | | 3 | Invalid Phone format. | System shows error message: “Phone cannot be blank or contain alphabetical characters and must be 9-11 digits”. | | 4 | Partner username already exist. | System shows error message: “This username already exist, please input another username”. |   **Relationships:** N/A.  **Business Rules:**   * Partner account’s password will be auto send back to partner via provided email. | | | |
| Table 20: USE CASE - UC\_AD006 - <Admin> Create account for Partner | | | |

* + - * 1. <Admin> Edit partner account (UC\_AD007)

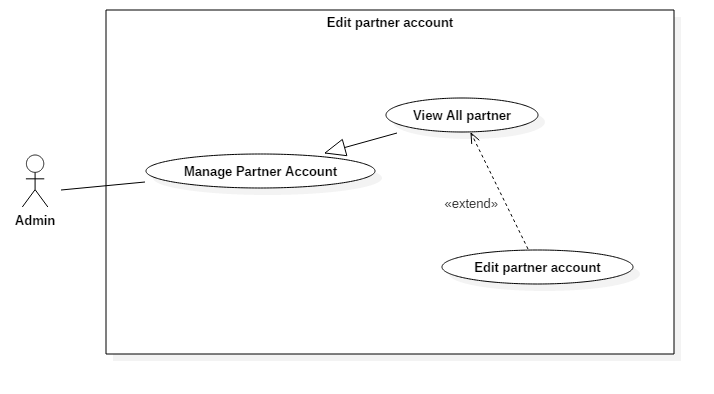


Figure 9: <Admin> Edit partner account (UC\_AD007)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_AD007** | | | |
| **Use Case No.** | UC\_AD007 | **Use Case Version** | 2.0 |
| **Use Case Name** | Edit partner account | | |
| **Author** | DanhLC | | |
| **Date** | 19/06/2018 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case allows actor to edit partner account information.   **Goal:**   * Successfully the information of specific partner account will be edited.   **Triggers:**   * Actor sends command to edit partner account.   **Preconditions:**   * Actor must login as Admin role.   **Post Conditions:**   * **Success:** Actor has to full fill information to edit partner account. * **Fail:** N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen, Actor click “manage partner account” | System requires display list of partner account as table. | | 2 | Actor chooses account they want to edit by click “detail” next to this account. | System will display a form. That form will full fill current information of this partner account. System requires information from admin:   * **Phone**: free text input, required, length 9-11 digits. * **Address**: free text input, required, length 8-500 characters. | | 3 | Actor click “save” button. | System validates information.  [Exception 1, 2, 3, 4, 5] | | 4 |  | System edits that partner account with inputted information to database. |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Invalid Email format | System shows error message: “Email cannot be blank and must be 8-40 characters long”. | | 2 | Invalid Phone format. | System shows error message: “Phone cannot be blank or contain alphabetical characters and must be 9-11 digits”. |   **Relationships:** N/A  **Business Rules:**   * We delete partner account by changing the status to disable. When status was change, user can’t use this account any more. | | | |
| Table 21: USE CASE - UC\_AD007 - <Admin> Edit partner account | | | |

* + - * 1. <Admin> Delete partner account (UC\_AD007)

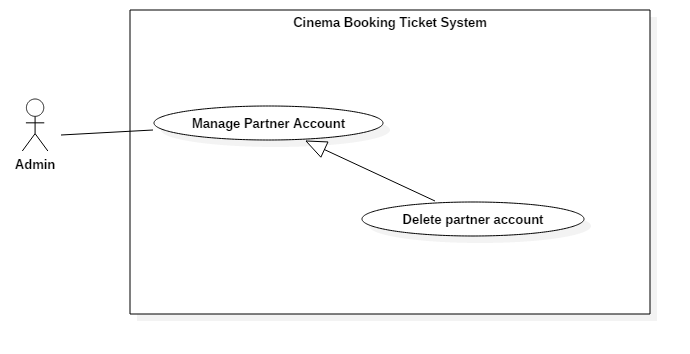


Figure 10: <Admin> Delete partner account (UC\_AD008)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_AD008** | | | |
| **Use Case No.** | UC\_AD008 | **Use Case Version** | 2.0 |
| **Use Case Name** | Delete partner account | | |
| **Author** | DanhLC | | |
| **Date** | 19/06/2018 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case allows actor to delete partner account that already exist to the system. When account was deleted, partner cannot login to the system any more.   **Goal:**   * Successfully this partner account cannot be used to login Partner.   **Triggers:**   * Actor sends command to delete partner account.   **Preconditions:**   * Actor must login as Admin role.   **Post Conditions:**   * **Success:** Action complete. * **Fail:** N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen, Actor click “Manage partner account” | System requires display list of partner account as table. | | 2 | Actor chooses account they want to delete and click “delete” next to this account. | System change status of this account to unavailable. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:** N/A | | | |
| Table 22: USE CASE - UC\_AD008 - <Admin> Delete partner account | | | |

* + - * 1. <Admin> View List Film (UC\_AD009)

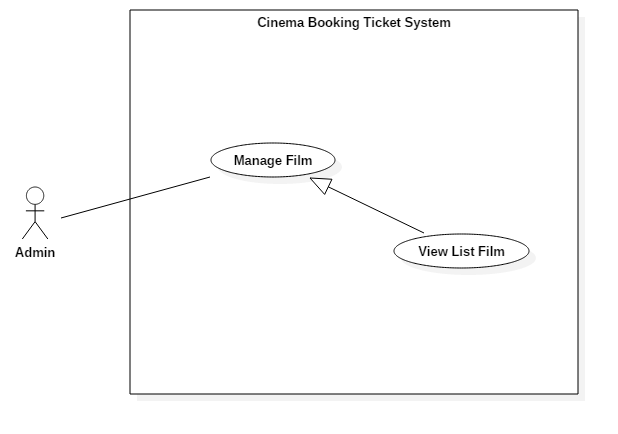


Figure 11: <Admin> View list film (UC\_AD009)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_AD009** | | | |
| **Use Case No.** | UC\_AD009 | **Use Case Version** | 2.0 |
| **Use Case Name** | View List Film | | |
| **Author** | DanhLC | | |
| **Date** | 19/06/2018 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case allows actor to view all list film that exist in the system.   **Goal:**   * Successfully Actor can view the list of film that exists in the system.   **Triggers:**   * Actor sends command to see all list film.   **Preconditions:**   * Actor must login as Admin role.   **Post Conditions:**   * **Success:** N/A * **Fail:** N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen, Actor click “Manage Film”  *[Alternative 1]* | System will display list of all film as table, each row of table will display following information:   * **Film name**: text, read only. * **imdb**: text, read only. * **Release date**: text, read only. * **Author**: text, read only. * **Film status**: text, read only. |   **Alternative Scenario:**  *[Alternative 1]*   |  |  |  | | --- | --- | --- | | Step. | Actor Action | System Response | | 1 | At home screen, Actor click “Manage Film” | System will display list of all film that existing in the system:   * **Film name**: text, read only. * **imdb**: text, read only. * **Release date**: text, read only. * **Author**: text, read only. * **Film status**: text, read only. |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:** N/A | | | |
| Table 23: USE CASE - UC\_AD009 - <Admin> View List Film | | | |

* + - * 1. <Admin> Add Film (UC\_AD010)

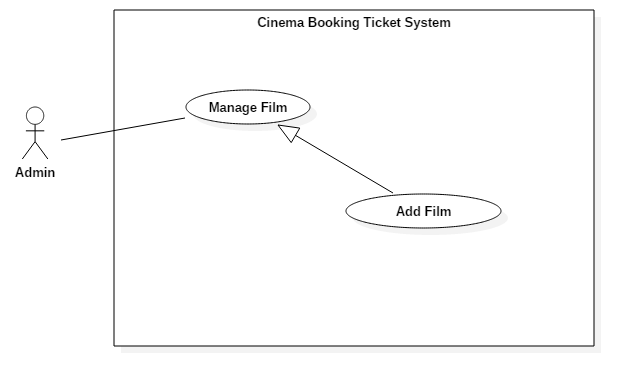


Figure 12: <Admin> Add Film(UC\_AD010)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_AD010** | | | |
| **Use Case No.** | UC\_AD010 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Film | | |
| **Author** | DanhLC | | |
| **Date** | 19/06/2018 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case allows actor to add new film to the system. From now, all the cinemas in the system can add schedule for this film.   **Goal:**   * Successfully Actor can add the film in to the system.   **Triggers:**   * Actor sends command to add new film.   **Preconditions:**   * Actor must login as Admin role.   **Post Conditions:**   * **Success:** Action Complete. * **Fail:** N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen, Actor click “Manage Film” | System will display manages film screen. | | 2 | Click “Add Film” button | Input form will display, request admin input following information:   * **Film Name**: hidden free text input, required, length 8–40 characters. * **Date release:** date input required. * **Restricted**: select from dropdown following digit (13,16,18) * **Film length**: digit number. * **Imdb**: decimal number required. The number is limit from 0 -> 10. * **Digital type**: select digital type from check box. * **Author**: free text input. * **Movie genre**: free text input. * **Film content**: free text input, limit to 1000 character. * **Actor list**: free text input. * **Countries**: free text input. * **Trailer link**: hidden free text input, required, length 8–255 characters. * **Poster picture:** hidden free text input, required, length 8–255 characters. * **Addition picture:** hidden free text input, required, length 8–255 characters. * **Film Status:** hidden text input. Select from dropdown list. Default is showing. | | 3 | Actor inputs information. |  | | 4 | Actor sends command to save.  [Alternative 1] | System validates information.  [Exception 1, 2, 3, 4] | | 5 |  | System adds a new film with inputted information to the database and immediately updates information for current screen film list. |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Invalid film name | System shows error message: “film name cannot be blank and must be 8-40 characters long”. | | 2 | Invalid release date | System shows error message: “release date cannot be blank”. | | 3 | Invalid poster picture | System shows error message: “Picture cannot be empty and must be in jpeg/jpg/png format”. | | 4 | Invalid addition picture | System shows error message: “Picture cannot be empty and must be in jpeg/jpg/png format”. |   **Relationships:** N/A  **Business Rules:** N/A | | | |
| Table 24: USE CASE - UC\_AD010 - <Admin> Add Film | | | |

* + - * 1. <Admin> Edit Film (UC\_AD011)

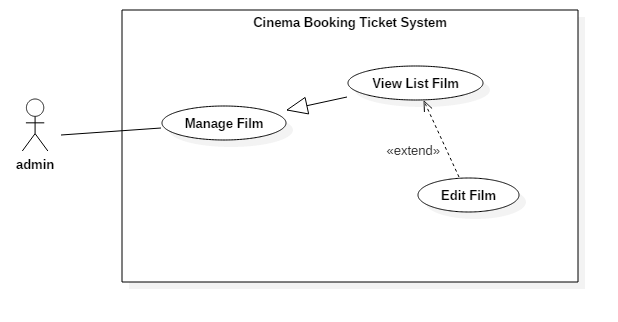


Figure 13: <Admin> Edit Film (UC\_AD011)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_AD011** | | | |
| **Use Case No.** | UC\_AD011 | **Use Case Version** | 2.0 |
| **Use Case Name** | Edit Film | | |
| **Author** | DanhLC | | |
| **Date** | 19/06/2018 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case allows actor to edit film that already exist in the system.   **Goal:**   * Successfully the film information will be edited.   **Triggers:**   * Actor sends command to edit film.   **Preconditions:**   * Actor must login as Admin role.   **Post Conditions:**   * **Success:** Action complete. * **Fail:** N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen, Actor click “Manage Film” | System will display manages film screen. | | 2 | Click “Detail” button | System will display a form. That form will full fill current information of selected film. System requires information from admin:   * **Film Name**: hidden free text input, required, length 8–40 characters. * **Date release:** date input required. * **Restricted**: select from dropdown following digit (13,16,18) * **Film length**: digit number. * **Imdb**: decimal number required. The number is limit from 0 -> 10. * **Digital type**: select digital type from check box. * **Author**: free text input. * **Movie genre**: free text input. * **Film content**: free text input, limit to 1000 character. * **Actor list**: free text input. * **Countries**: free text input. * **Trailer link**: hidden free text input, required, length 8–255 characters. * **Poster picture:** hidden free text input, required, length 8–255 characters. * **Addition picture:** hidden free text input, required, length 8–255 characters. * **Film Status:** hidden text input. Select from dropdown list. Default is showing. | | 3 | Actor inputs information. |  | | 4 | Actor sends command to save. | System validates information.  [Exception 1, 2, 3, 4] | | 5 |  | System edits the film that was selected with inputted information, save it to the database and immediately update information for current screen film list. |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Invalid film name | System shows error message: “film name cannot be blank and must be 8-40 characters long”. | | 2 | Invalid release date | System shows error message: “release date cannot be blank”. | | 3 | Invalid poster picture | System shows error message: “Picture cannot be empty and must be in jpeg/jpg/png format”. | | 4 | Invalid addition picture | System shows error message: “Picture cannot be empty and must be in jpeg/jpg/png format”. |   **Relationships:** N/A  **Business Rules:** N/A | | | |
| Table 25: USE CASE - UC\_AD011 - <Admin> Edit Film | | | |

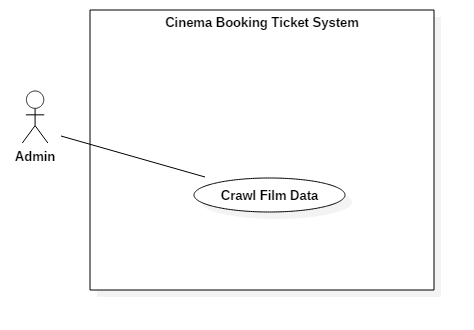
* + - * 1. <Admin> Crawl Film Data (UC\_AD012) 

Figure 14 <Admin> Crawl film data (UC\_AD012)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_AD012** | | | |
| **Use Case No.** | UC\_AD012 | **Use Case Version** | 2.0 |
| **Use Case Name** | Crawl Film Data | | |
| **Author** | DanhLC | | |
| **Date** | 19/06/2018 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case allows actor to crawl film data from web side to save to the system.   **Goal:**   * Successfully the film information that user selected will be added to database.   **Triggers:**   * Actor sends command to crawl film.   **Preconditions:**   * Actor must login as Admin role.   **Post Conditions:**   * **Success:** N/A * **Fail:** N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen, Actor click “Crawl Data” | System will display crawl film data screen. | | 2 | Choose month for crawl data | System will automatically crawl film data of this month from the internet and display film on the screen. | | 3 | Actors click add film | The film will added to the list | | 4 | Actor click view added list | The list of all film user has added will show | | 5 | Actor click “save” button | System will save all selected film to database.  [Exception 1] |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | The film already exist! | System shows error message: “System detect that this film already exist”. |   **Relationships:** N/A  **Business Rules:**   * Film crawl just from current month to the last month of this year. * User must review film data first to see if it already exist in the system before decide to add it. | | | |
| Table 26: <admin> Crawl Film Data (UC\_AD012) | | | |

* + - 1. <Partner> Overview Use Case

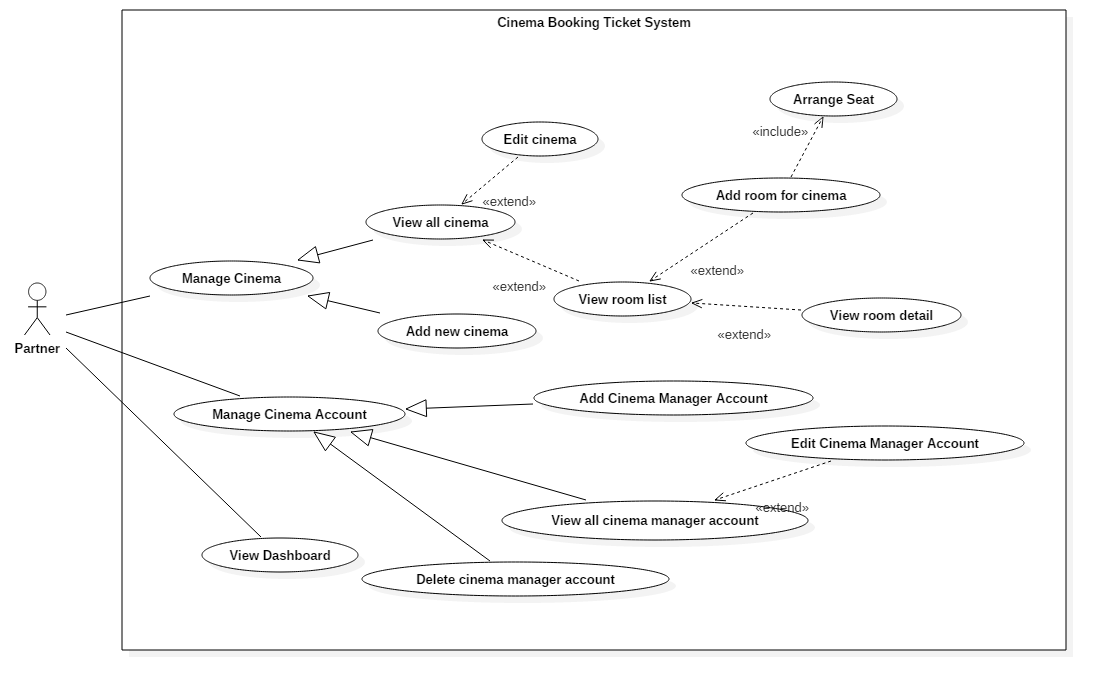


Figure 15: <User> Overview Use Case

* + - * 1. <Partner>View All Cinema Manager (UC\_P002)

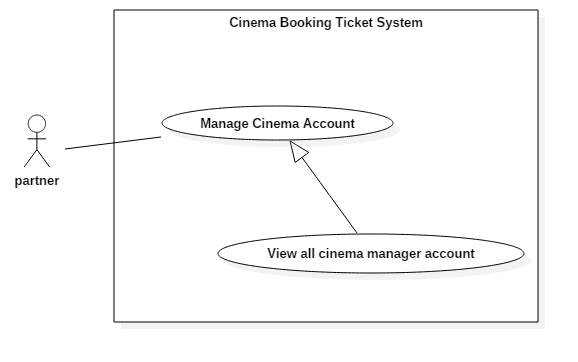


Figure 16: <Partner> View all cinema manager account (UC\_P002)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_P002** | | | |
| **Use Case No.** | UC\_P002 | **Use Case Version** | 2.0 |
| **Use Case Name** | View all cinema manager account | | |
| **Author** | DanhLC | | |
| **Date** | 20/06/2018 | **Priority** | Normal |
| **Actor:**   * Partner   **Summary:**   * This use case allows actor to view list of all cinema manager as table form.   **Goal:**   * Actor can view list of all cinema manager.   **Triggers:**   * Actor sends command to view list of all cinema manager.   **Preconditions:**   * Actor must login as Partner.   **Post Conditions:**   * **Success:** The action is completed. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen, Actor click “Manage Cinema Manager Account” | System will display list of all cinema manager as table, each row of table will display following information:   * **Account username**: text, read only. * **Cinema name**: text, read only. * **phone**: text, read only. * **email**: text, read only. |   **Alternative Scenario: N/A**  **Exceptions: N/A**  **Relationships: N/A**  **Business Rules:** N/A | | | |
| Table 27: USE CASE - UC\_P002 - <Partner> View all cinema manager account | | | |

* + - * 1. <Partner> Add Cinema Manager Account (UC\_P003)

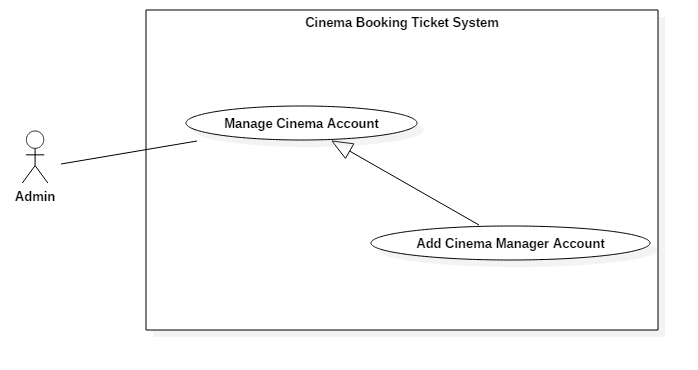


Figure 17: <Partner> Add Cinema Manager Account (UC\_P003)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_P003** | | | |
| **Use Case No.** | UC\_P003 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add Cinema Manager Account | | |
| **Author** | DanhLC | | |
| **Date** | 19/06/2018 | **Priority** | Normal |
| **Actor:**   * Partner   **Summary:**   * This use case allows actor to add new cinema manager account. Cinema manager is considered as child of partner. If partner manage a group of cinemas. Cinema manager will manage each cinema in this group cinema.   **Goal:**   * New cinema manager account will be added to the system.   **Triggers:**   * Actor sends command to add new cinema manager account.   **Preconditions:**   * Actor must login as Partner role.   **Post Conditions:**   * **Success:** Actor must full fill the information in right format and click “save”. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen, Actor click “Manage Cinema Manager Account” | System will display manages cinema manager account screen. | | 2 | Click “Add Cinema Manager Account” button | System will have required input following field:   * **Cinema manager username:** free text input, required, length 8–40 characters. * **Cinema manager password:** free text input, required, length 8–40 characters. * **Phone**: free text input, required, length 9-11 digits. * **Email**: free text input, required, length 8-40 characters. * **CinemaId**: select from dropdown list. Dropdown list will contain all cinema of current group cinema. | | 3 | Click button “add”. | System validates information.  [Exception 1, 2, 3, 4, 5] | | 4 |  | System adds new cinema manager account to database. |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Invalid cinema manager username format. | System shows error message: “cinema manager username cannot be blank and must be 8-40 characters long”. | | 2 | Invalid cinema manager password format | System shows error message: “password cannot be blank and must be 8-40 characters long”. | | 3 | Invalid email password format | System shows error message: “Email cannot be blank and must be 8-40 characters long”. | | 4 | Invalid Phone format. | System shows error message: “Phone cannot be blank or contain alphabetical characters and must be 9-11 digits, ”. | | 5 | Partner username already exist | System shows error message: “This username already exists, please input another username”. |   **Relationships:** N/A  **Business Rules:**   * After create, the account information will be automatic send back to cinema manager via email. | | | |
| Table 28: USE CASE - UC\_P003 <Partner> Add Cinema Manager Account | | | |

* + - * 1. <Partner>Delete Cinema Manager Account (UC\_P004)

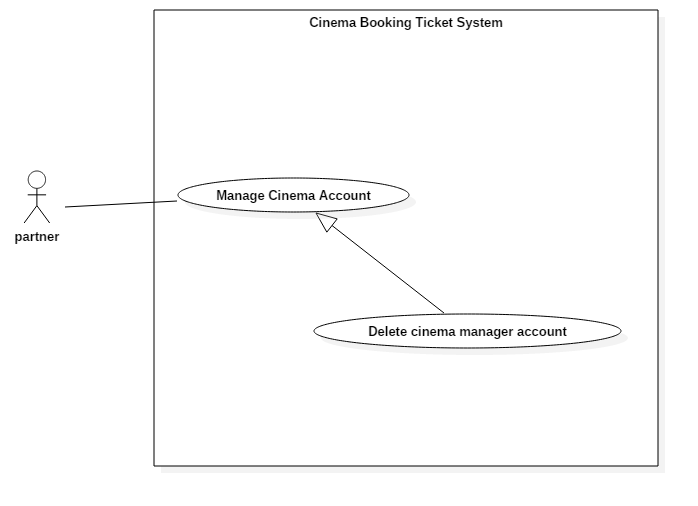


Figure 18: <Partner>Delete Cinema Manager Account (UC\_P004)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_P004** | | | |
| **Use Case No.** | UC\_P004 | **Use Case Version** | 2.0 |
| **Use Case Name** | Delete cinema manager account | | |
| **Author** | DanhLC | | |
| **Date** | 20/06/2018 | **Priority** | Normal |
| **Actor:**   * Partner   **Summary:**   * This use case allows actor to delete cinema manger account. When cinema manager account has been deleted. Cinema manager cannot login with this account into the system. Partner also cannot see this account when click “view all cinema manager account”.   **Goal:**   * Delete cinema manager account.   **Triggers:**   * Actor sends command to delete cinema manager account.   **Preconditions:**   * Actor must login as partner role.   **Post Conditions:**   * **Success:** Action completed. * **Fail:** N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen, Actor click “Manage Cinema Manager Account” | System will display list of cinema manager account in table form. | | 2 | Actor chooses cinema manager account want to delete and click “delete” next to this account. | System change status of this account to unavailable. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:** N/A | | | |
| Table 29: USE CASE - UC\_P004 <Partner>Delete Cinema Manager Account | | | |

* + - * 1. <Partner> Edit Cinema Manager Account (UC\_P005)

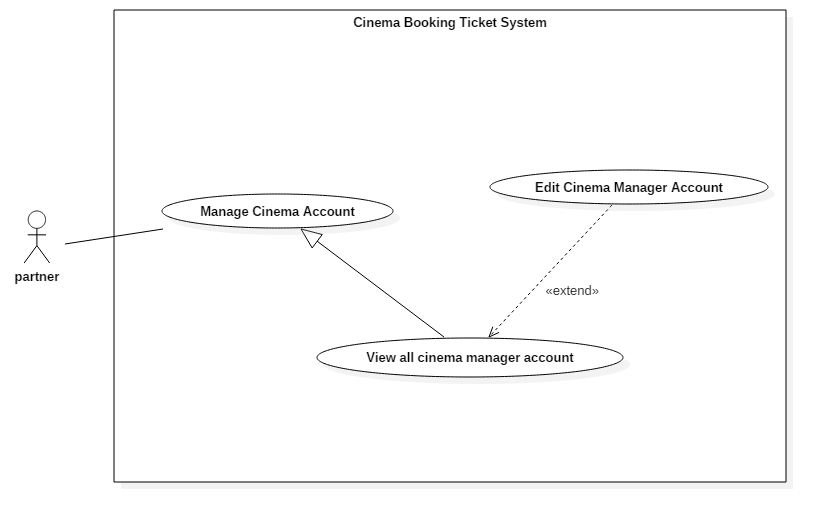


Figure 19: <Partner> Edit Cinema Manager Account (UC\_P005)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_P005** | | | |
| **Use Case No.** | UC\_P005 | **Use Case Version** | 2.0 |
| **Use Case Name** | Edit Cinema Manager Account | | |
| **Author** | DanhLC | | |
| **Date** | 20/06/2018 | **Priority** | Normal |
| **Actor:**   * Partner   **Summary:**   * This use case allows actor to edit manager account information.   **Goal:**   * Update new information for selected cinema manager account.   **Triggers:**   * Actor sends command to edit cinema manager account.   **Preconditions:**   * Actor must login as Partner role.   **Post Conditions:**   * **Success:** Partner fills new information of cinema manager in right format. * **Fail:** System will display error messages below.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen, Actor click “Manage Cinema Manager Account” | System will display list of cinema manager account in table form. | | 2 | Click “Detail” button | System will show form fill current cinema manager account that was selected. System allow actor edit following field of this form:   * **Cinema manager password:** free text input, required, length 8–40 characters. * **Phone**: free text input, required, length 9-11 digits. * **Email**: free text input, required, length 8-40 characters. * **CinemaId**: select from dropdown list. Dropdown list will contain all cinema of current group cinema. | | 3 | Click button “save”. | System validates information.  [Exception 1, 2, 3] | | 4 |  | System updates new information for the selected cinema manager account to database. |   **Alternative Scenario: N/A**  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Invalid cinema manager password format | System shows error message: “password cannot be blank and must be 8-40 characters long”. | | 2 | Invalid email password format | System shows error message: “Email cannot be blank and must be 8-40 characters long”. | | 3 | Invalid Phone format. | System shows error message: “Phone cannot be blank or contain alphabetical characters and must be 9-11 digits, ”. |   **Relationships:** N/A  **Business Rules:** After created, the account information will be automatic sent back to cinema manager via email. | | | |
| Table 30: USE CASE - UC\_P005 <Partner> Edit Cinema Manager Account | | | |

* + - * 1. <Partner> View all cinema (UC\_P006)

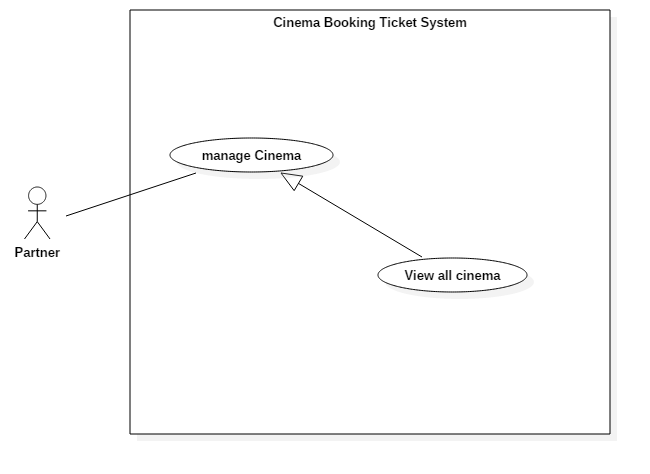


Figure 20: <Partner> View all cinema (UC\_P006)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_P006** | | | |
| **Use Case No.** | UC\_P006 | **Use Case Version** | 2.0 |
| **Use Case Name** | View All Cinema | | |
| **Author** | DanhLC | | |
| **Date** | 20/06/2018 | **Priority** | Normal |
| **Actor:**   * User   **Summary:**   * This use case allows actor to view all list of cinemas that belong to this group cinema. Each cinema will manage by cinema manager.   **Goal:**   * Actor can view all cinema belong to their cinema group.   **Triggers:**   * Actor sends command to view all cinema group.   **Preconditions:**   * Actor must login as Partner role.   **Post Conditions:**   * **Success:** Action completed. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen, Actor click “Manage Cinema” | System will display list of all cinema as table, each row of table will display following information:   * **Cinema name**: text, read only. * **Cinema address**: text, read only. * **phone**: text, read only. * **email**: text, read only. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:** N/A | | | |
| Table 31: USE CASE - UC\_P006 <Partner> View all cinema | | | |

* + - * 1. <Partner> Arrange seat (UC\_P007)

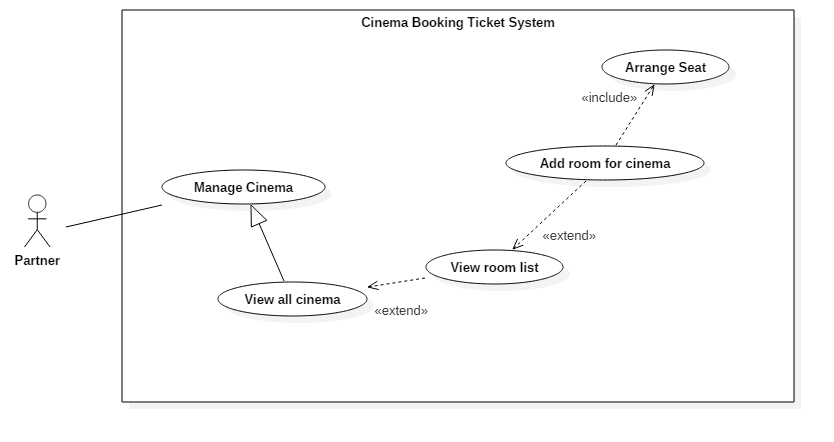


Figure 21: <Partner> Arrange seat (UC\_P007)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_P007** | | | |
| **Use Case No.** | UC\_P007 | **Use Case Version** | 2.0 |
| **Use Case Name** | Arrange seat | | |
| **Author** | DanhLC | | |
| **Date** | 20/06/2018 | **Priority** | Normal |
| **Actor:**   * Partner   **Summary:**   * When partner create a cinema. They may want to add room to their cinema. And when a room is created, we allow partner to arrange seat for this room. The strategy for this is as following. First of all, when create room, we ask them for input room capacity and system will automatically calculate the dimension x and y of this room. We will create a room template match with what partner has inputted above. The room will contain no seat available. After that, partner can add seat to this template. If there is no seat added in an area, this area will become an empty area.   **Goal:**   * To arrange seat for a room of specific cinema.   **Triggers:**   * Actor sends command to add arrange seat for cinema.   **Preconditions:**   * Actor must login as Partner.   **Post Conditions:**   * **Success:** The action is completed. * **Fail:** System will display error messages below.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Actor chooses seat that they want to add on the seat matrix template.  [Alternative 1] | A seat was added at the position that they click. The color of this seat will change from gray to orange meaning available. | | 2 | Partner continues repeat step 1 until the number of seat selected equal to capacity user has inputted. |  | | 3 | Partner click “save” button | System validates information.  [Exception 1] | | 4 |  | System will save the matrix of seat in to the database. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | Step. | Actor Action | System Response | | 1 | Actor chooses seat that they want to remove on the seat matrix template. | A seat was removed at the position that they click. The color of this seat will change from orange to gray meaning unavailable. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Number of seat selected not full fill capacity user has inputted. | System shows message: “You not select enough seat.” |   **Relationships:**   * Include with add room to cinema: when user want to add room to cinema they must to arrange seat for it.   **Business Rules:**   * Number of seat selected must full fill capacity of the room. | | | |
| Table 32: USE CASE - UC\_P007 <Partner> Arrange seat | | | |

* + - * 1. <Partner> Add Room for Cinema (UC\_P008)

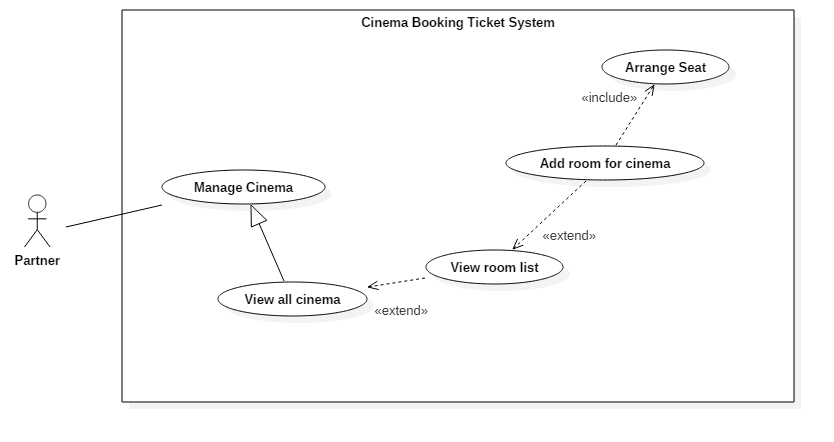


Figure 22: <Partner> Add Room for Cinema (UC\_P008)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_P008** | | | |
| **Use Case No.** | UC\_P008 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add room for cinema | | |
| **Author** | DanhLC | | |
| **Date** | 20/06/2018 | **Priority** | Normal |
| **Actor:**   * Partner   **Summary:**   * This use case allows actor to add room to specific cinema. When list user show, they can extend to add room for this cinema.   **Goal:**   * This use case help actor to add a room to specific cinema.   **Triggers:**   * Actor sends command to add room for cinema.   **Preconditions:**   * Actor must login as Partner.   **Post Conditions:**   * **Success:** The action is completed. * **Fail:** System will display error messages below.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen, actor click button “manage cinema”. | System display screen for manage cinema. The list of cinemas will show as a table. | | 2 | Select a row represent for a cinema and click “add room” button at the end of this row. | System will show form for add room and required following input:   * **Capacity:** free number input, value is from 80 to 300. * **Room’s width**: free number input, required, value is from 5 to 20. * **Room’s height**: free number input, required, value is from 5 to 20. | | 3 | Actor sends command to save. | System validates information.  [Exception 1, 2, 3, 4] | | 4 |  | System adds a new room for the selected cinema with inputted information. |   **Alternative Scenario: N/A**  **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Cause | System Response | | 1 | Invalid capacity input. | System shows message: “capacity must be number and value is between 80 and 300”. | | 2 | Invalid Room’s width input | System shows message: “dimension X must be number and value is between 5 and 20” | | 3 | Invalid Room’s height input | System shows message: “dimension X must be number and value is between 5 and 20” | | 4 | Capacity input is less than multiple of width and height. | System shows message: “capacity must less than multiple of width and height”. |   **Relationships:**   * Extend from view all cinemas: when user wants to show list of cinemas, they can extend for add room for this cinema. * Include with arrange seat. When a room is created they must arrange seat for it.   **Business Rules:**   * Once you click save, you cannot edit seat position any more. * User must select full seat that they has inputted. | | | |
| Table 33: USE CASE - UC\_P008 <Partner> Add Room for Cinema | | | |

* + - * 1. <Partner> Add New Cinema (UC\_P009)



Figure 23: <Partner> Add New Cinema (UC\_P009)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_P009** | | | |
| **Use Case No.** | UC\_P009 | **Use Case Version** | 2.0 |
| **Use Case Name** | Add new cinema | | |
| **Author** | DanhLC | | |
| **Date** | 20/06/2018 | **Priority** | Normal |
| **Actor:**   * Partner   **Summary:**   * This use case allows actor to add new cinema for current group cinema.   **Goal:**   * To add new cinema for current group cinema.   **Triggers:**   * Actor sends command to add new cinema.   **Preconditions:**   * Actor must login as Partner role.   **Post Conditions:**   * **Success:** The action is completed. * **Fail:** N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home, click button “mange cinema”. | System display list of cinemas in table form. | | 2 | Click “add new” button. | System will have required input following field:   * **Cinema name:** free text input, required, length 8–40 characters. * **Cinema address:** free text input, required, length 8–255 characters. * **Phone**: free text input, required, length 9-11 digits. * **Email**: free text input, required, length 8-40 characters. * **Introduction**: free text input, required, length 0-1000 characters. * **Profile picture:** hidden free text input, required, length 8–255 characters. | | 3 | Actor inputs information. |  | | 4 | Actor sends command to save. | System validates information.  [Exception 1, 2, 3, 4,5] | | 5 |  | System adds a new cinema with inputted information to the database and immediately updates information for current screen cinema list. |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Invalid Address format. | System shows error message: “Address cannot be blank and must be 8-40 characters long”. | | 2 | Invalid Email format | System shows error message: “Email cannot be blank and must be 8-40 characters long and must match email format”. | | 3 | Invalid Phone format. | System shows error message: “Phone cannot be blank or contain alphabetical characters and must be 9-11 digits, ”. | | 4 | Invalid Name format. | System shows error message: “Name cannot be blank and must be 8-40 characters long”. | | 5 | Invalid profile picture | System shows error message: “Picture cannot be empty and must be in jpeg/jpg/png format”. |   **Relationships: N/A**  **Business Rules:** N/A | | | |
| Table 34: USE CASE - UC\_P009 <Partner> Add New Cinema | | | |

* + - * 1. <Partner> Edit cinema (UC\_P010)

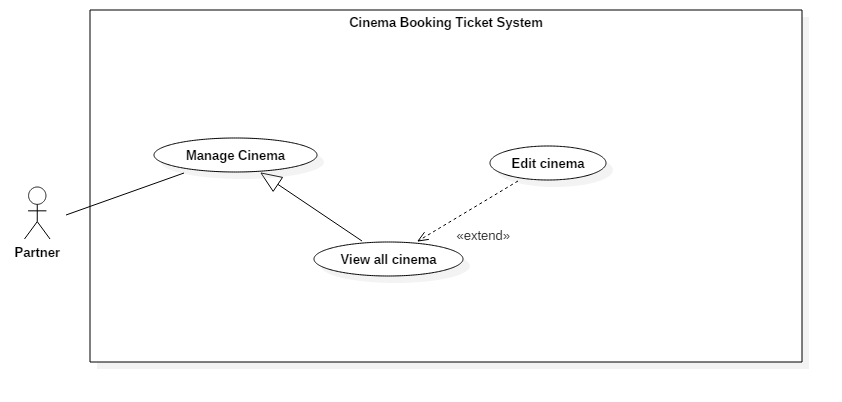


Figure 24: <Partner> Edit cinema (UC\_P010)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_P010** | | | |
| **Use Case No.** | UC\_P010 | **Use Case Version** | 2.0 |
| **Use Case Name** | Edit cinema | | |
| **Author** | DanhLC | | |
| **Date** | 20/06/2018 | **Priority** | Normal |
| **Actor:**   * Partner   **Summary:**   * This use case allows actor to edit cinema information.   **Goal:**   * Update new information for selected cinema.   **Triggers:**   * Actor sends command to edit cinema information.   **Preconditions:**   * Actor must login as Partner.   **Post Conditions:**   * **Success:** The action is completed. * **Fail:** Show error message as below.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen, Actor click “Manage Cinema” | System will display manages cinema screen. | | 2 | Click “Detail” button | System will display a form. That form will full fill current information of selected cinema. System requires information from partner:   * **Cinema name:** free text input, required, length 8–40 characters. * **Cinema address:** free text input, required, length 8–255 characters. * **Phone**: free text input, required, length 9-11 digits. * **Email**: free text input, required, length 8-40 characters. * **Introduction**: free text input, required, length 0-1000 characters. * **Profile picture:** hidden free text input, required, length 8–255 characters. | | 3 | Actor inputs information. |  | | 4 | Actor sends command to save. | System validates information.  [Exception 1, 2, 3, 4] | | 5 |  | System updates information for selected cinema with inputted information to the database and immediately updates information for current screen cinema list. |   **Alternative Scenario:**  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Invalid Address format. | System shows error message: “Address cannot be blank and must be 8-40 characters long”. | | 2 | Invalid Email format | System shows error message: “Email cannot be blank and must be 8-40 characters long and must match email format”. | | 3 | Invalid Phone format. | System shows error message: “Phone cannot be blank or contain alphabetical characters and must be 9-11 digits, ”. | | 4 | Invalid Name format. | System shows error message: “Name cannot be blank and must be 8-40 characters long”. | | 5 | Invalid profile picture | System shows error message: “Picture cannot be empty and must be in jpeg/jpg/png format”. |   **Relationships:** N/A  **Business Rules:** N/A | | | |
| Table 35: USE CASE - UC\_P010 <Partner> Edit cinema | | | |

* + - * 1. <Partner> View Room List (UC\_P011)

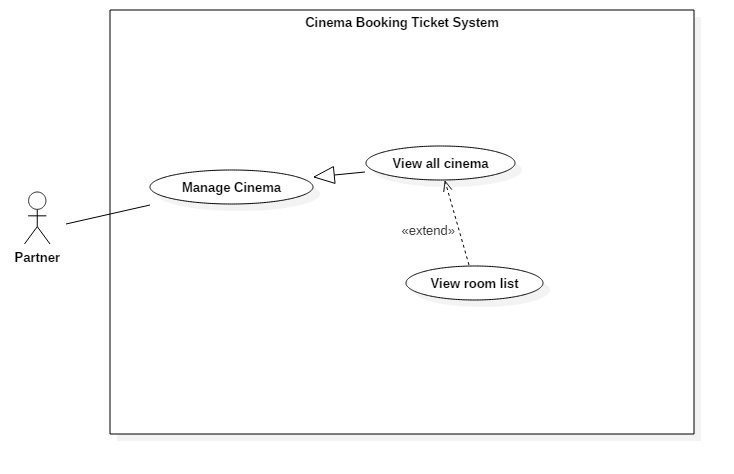


Figure 25: <Partner> View Room List (UC\_P011)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_P011** | | | |
| **Use Case No.** | UC\_P011 | **Use Case Version** | 2.0 |
| **Use Case Name** | View room list | | |
| **Author** | DanhLC | | |
| **Date** | 20/06/2018 | **Priority** | Normal |
| **Actor:**   * Partner   **Summary:**   * This use case allows actor view all room of specific cinema.   **Goal:**   * View all room of specific cinema.   **Triggers:**   * Actor sends command to view all room.   **Preconditions:**   * Actor must login as Partner.   **Post Conditions:**   * **Success:** The action is completed * **Fail:** N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen, Actor click “Manage Cinema” | System will display list of all cinema as table. | | 2 | Actor click “detail” button at the end of each row | System show all rooms belong to this cinema as table format. This table will stay below cinema list table. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:** N/A | | | |
| Table 36: USE CASE - UC\_P011 <Partner> View Room List | | | |

* + - * 1. <Partner> View Room Detail (UC\_P012)

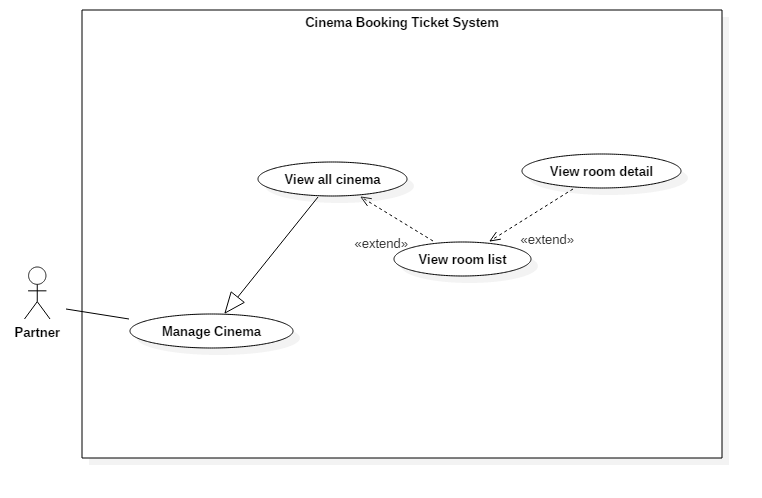


Figure 26: <Partner> View Room Detail (UC\_P012)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_P012** | | | |
| **Use Case No.** | UC\_P012 | **Use Case Version** | 2.0 |
| **Use Case Name** | View room detail | | |
| **Author** | DanhLC | | |
| **Date** | 20/06/2018 | **Priority** | Normal |
| **Actor:**   * Partner   **Summary:**   * This use case allows actor view all room of specific cinema.   **Goal:**   * View all room of specific cinema.   **Triggers:**   * Actor sends command to view all room.   **Preconditions:**   * Actor must login as Partner.   **Post Conditions:**   * **Success:** The action is completed * **Fail:** N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen, Actor click “Manage Cinema” | System will display list of all cinema as table. | | 2 | Actor click “detail” button at the end of each row of cinema table. | System show all rooms belongs to this cinema as table format. This table will stay below cinema list table. | | 3 | Actor click “detail” button at the end of each row of room table. | System gives the specific view about the room which user has added before. | | 4 | Actor click “x” button | Go back to manage cinema screen. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:** N/A | | | |
| Table 37: USE CASE - UC\_P012 <Partner> View Room Detail | | | |

* + - 1. <User> Overview Use Case

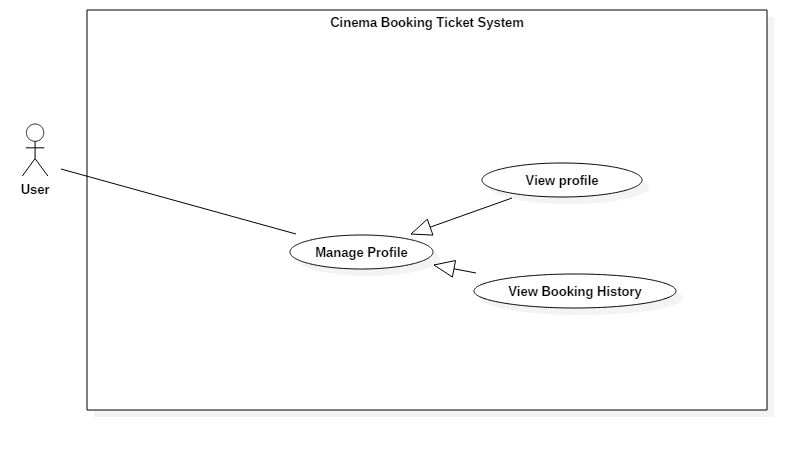


Figure 27: <User> Overview Use Case

* + - * 1. <User> View Profile (UC\_U002)

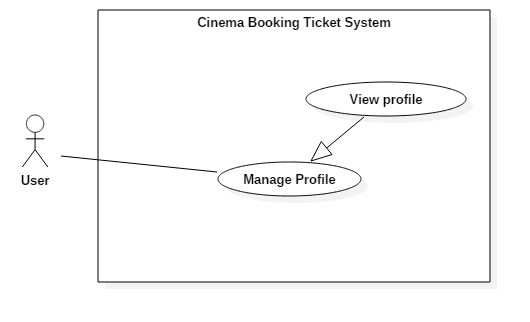
****

Figure 28: <User> View Profile (UC\_U002)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_U002** | | | |
| **Use Case No.** | UC\_U002 | **Use Case Version** | 2.0 |
| **Use Case Name** | View Profile | | |
| **Author** | DanhLC | | |
| **Date** | 20/06/2018 | **Priority** | Normal |
| **Actor:**   * User   **Summary:**   * This use case allows actor to view their information.   **Goal:**   * View profile information.   **Triggers:**   * Actor sends command to view profile.   **Preconditions:**   * Actor must login as User.   **Post Conditions:**   * **Success:** The action is completed. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home screen, Actor click “view profile” | System will display their profile information. The information will include:   * **username**: text, read only. * **phone** text, read only. * **email**: text, read only. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:** N/A | | | |
| Table 38: USE CASE - UC\_U002 <User> View Profile | | | |

* + - * 1. <User> View Booking History (UC\_U004)

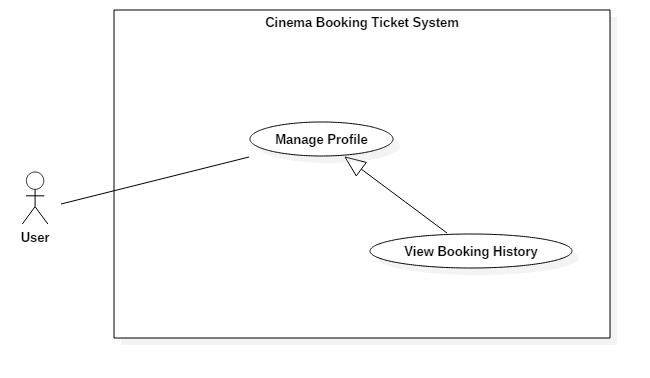


Figure 29: <User> View Booking History (UC\_U004)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_U004** | | | |
| **Use Case No.** | UC\_ | **Use Case Version** | 2.0 |
| **Use Case Name** | View Booking History | | |
| **Author** | DanhLC | | |
| **Date** | 20/06/2018 | **Priority** | Normal |
| **Actor:**   * User   **Summary:**   * This use case allows actor to view their booking history.   **Goal:**   * Actor can view their booking history.   **Triggers:**   * Actor sends command to view booking history.   **Preconditions:**   * Actor must login as User.   **Post Conditions:**   * **Success:** The action is completed. * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Actor goes to profile view. | System display profile screen. | | 2 | Actor click “view booking history” button. | System will display their booking history. The information will display in table form. The information will include:   * **Booking date**: text, read only. * **Film Name:** text, read only. * **Ticket quantity:** text, read only. * **Total amount**: text, read only. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:** N/A | | | |
| Table 39:USE CASE - UC\_U004 <User> View Booking History | | | |

* + - 1. <Customer> Overview Use Case

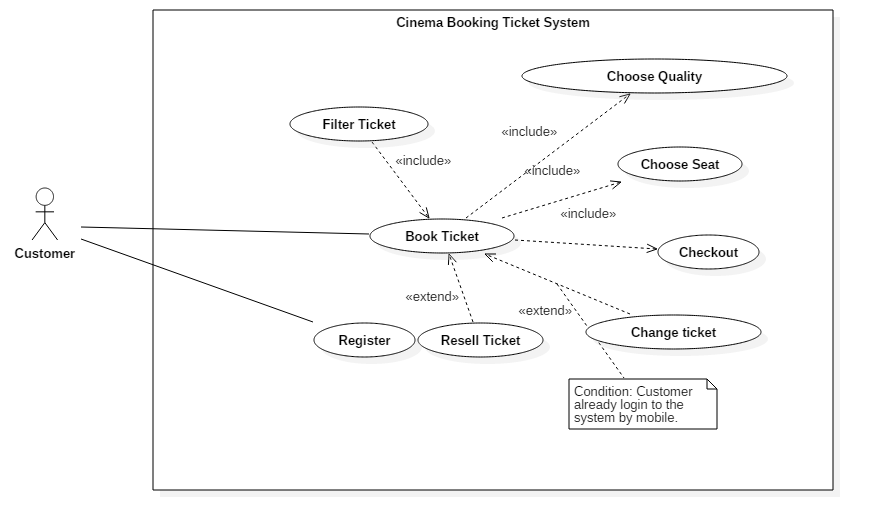


Figure 30: <Customer> Overview Use Case

* + - * 1. <Customer> Register (UC\_C001)

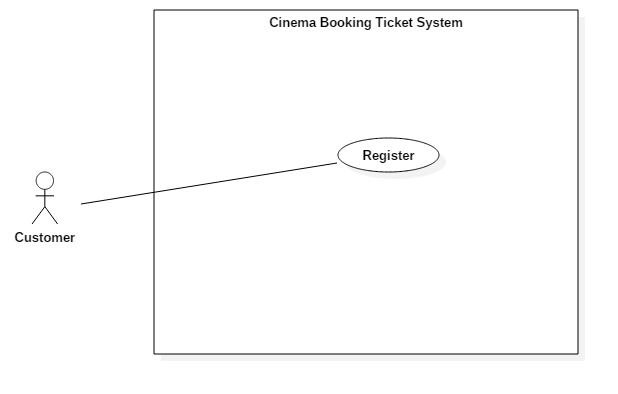


Figure 31: <Customer> Register (UC\_C001)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_C001** | | | |
| **Use Case No.** | UC\_C001 | **Use Case Version** | 2.0 |
| **Use Case Name** | Register | | |
| **Author** | DanhLC | | |
| **Date** | 17/01/2016 | **Priority** | Normal |
| **Actor:**   * Customer   **Summary:**   * This use case helps Customer registers a new account.   **Goal:**   * Actor can create a new account in the system.   **Triggers:**   * Actor sends sign up command.   **Preconditions:**   * N/A   **Post Conditions:**   * **Success:** Account is created successfully * **Fail:** Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Actor goes to sign up view. | System requires some information from actor:  - **Username**: free text input, required, length 8 - 40  **- Email**: free text input, required, length 8 – 40, regex: /\A([\w+\-].?)+@[a-z\d\-]+(\.[a-z]+)\*\.[a-z]+\z/i - **Password**: free text input, required, length 8 - 40  -**Confirm** **password**: free text input, required, length 8 - 40 | | 2 | Actor inputs information |  | | 3 | Actor sends sign up command | System validates inputted information.  [Exception 1, 2, 3] | | 4 |  | System creates an account with Actor’s email |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | username already exists in the system | System shows error message “Account already exists”. | | 2 | Invalid username format. | System shows error message: “Username cannot blank and must be 8 – 40 character longs”. | | 3 | Invalid email format | System shows error message: “Not a valid email format”. | | 4 | Invalid password format | System shows error message: “Password cannot be blank and must be 8 – 40 characters long”. | | 5 | Confirm password does not match. | System shows error message: “These passwords do not match”. |   **Relationships:** N/A  **Business Rules:**   * After sign in to the system, actor will be redirected to home page. | | | |
| Table 40: USE CASE - UC\_C001 <Customer> Register | | | |

* + - * 1. <Customer> Find Schedule (UC\_C002)

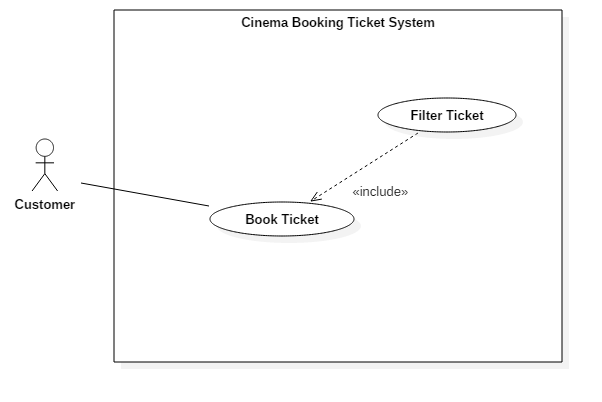


Figure 32: <Customer> Filter Ticket (UC\_C002)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_C002** | | | |
| **Use Case No.** | UC\_C002 | **Use Case Version** | 2.0 |
| **Use Case Name** | Filter Ticket | | |
| **Author** | DanhLC | | |
| **Date** | 20/06/2018 | **Priority** | High |
| **Actor:**   * Customer   **Summary:**   * This use case helps Actor to filter ticket by film, by group cinema, by time and date. After that user can easy choose the ticket that they want to buy.   **Goal:**   * Actor can easily choose suitable ticket.   **Triggers:**   * Actor can send request to filter ticket.   **Preconditions: N/A**  **Post Conditions:**   * **Success:** The action is completed * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Customer chooses the film that they want to buy ticket. | System goes to detail film page. At the end of the page will display quick navigate to select. | | 2 | Click on to group cinema on quick navigate. | System will display all cinema of this group cinema on quick navigate. | | 3 | Click on cinema on list cinema response by previous step. | System recommend all Show Time filter by current film, cinema. | | 4 | Click on date navigate | System recommend all Show Time filter by current film, cinema, date selected. |   **Alternative Scenario:**  *[Alternative 1]*   |  |  |  | | --- | --- | --- | | Step. | Actor Action | System Response | | 1 | In the quick filter at home page. User chooses the group cinema. | System will display all cinema of this group cinema on quick navigate. | | 2 | Click on cinema on list cinema response by previous step. | System recommends all Show Time and film filter by cinema of current day. |   **Exceptions:**  **Relationships:** Include with book ticket.  **Business Rules:** | | | |
| Table 41: USE CASE - UC\_C002 <Customer> Filter Ticket | | | |

* + - * 1. <Customer>Choose type of seat and quality (UC\_C003)

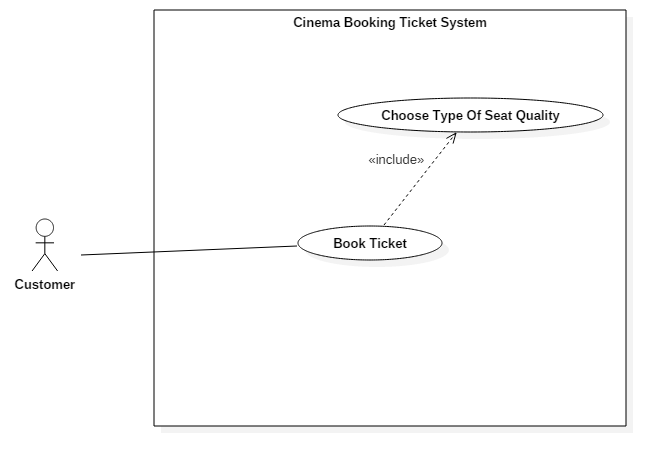


Figure 33: <Customer>Choose type of seat and quality (UC\_C003)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_C003** | | | |
| **Use Case No.** | UC\_C003 | **Use Case Version** | 2.0 |
| **Use Case Name** | Choose type of seat quality | | |
| **Author** | DanhLC | | |
| **Date** | 20/06/2018 | **Priority** | normal |
| **Actor:**   * Staff   **Summary:**   * This use case helps Actor to choose the type of seat and number of ticket that they want to buy.   **Goal:**   * Actor can choose the type of seat and number of ticket that they want to buy.   **Triggers:**   * Actor sends filter command to choose ticket by click in show time of the ticket that they want to buy.   **Preconditions:**   * Actor must choose the ticket that they want to buy success.   **Post Conditions:**   * **Success:** The action is completed * **Fail:** N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Actor click on the show time of the ticket that they want to buy | System will show the screen with all type of seat. | | 2 | Actor click on “+” / “-“ button next to type of seat that they want to buy | Increase/decrease the ticket quality with that type of seat. The ticket quality for all type of seat will limit to 10.  [Exception 1] |   **Alternative Scenario: N/A**  **Exceptions: N/A**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Number of seat is more than 10. | System shows error message: “Bạn chỉ được chọn 10 vé”. | | 2 | Number of seat is equal 0 | System will automatic disable “chọn ghế” button. So they cannot go to the next step to book ticket. |   **Relationships:** Include with book ticket.  **Business Rules:** | | | |
| Table 42: USE CASE - UC\_C003 <Customer> Choose type of seat and quality | | | |

* + - * 1. <Customer> Choose Seat (UC\_C004)

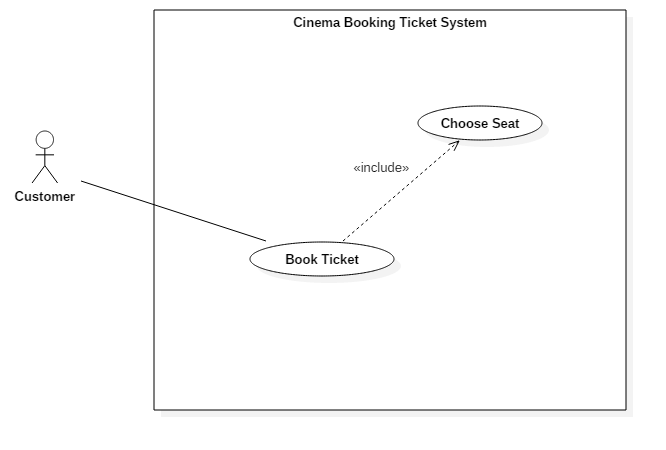


Figure 34: <Customer> Choose Seat (UC\_C004)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_C004** | | | |
| **Use Case No.** | UC\_C004 | **Use Case Version** | 2.0 |
| **Use Case Name** | Choose Seat | | |
| **Author** | DanhLC | | |
| **Date** | 20/06/2018 | **Priority** | High |
| **Actor:**   * Customer   **Summary:**   * This use case helps actor can choose their seat position in the room that they will see the film.   **Goal:**   * Actor can choose their seat position in the room that they will see the film.   **Triggers:**   * Actor sends choose seat command.   **Preconditions:**   * Users must choose type of seat quality success.   **Post Conditions:**   * **Success:** The action is completed. * **Fail:** Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | After choose type of seat and ticket quality success. Actor now moves to choose seat screen. | 2D room cinema will show. This screen will allow user to choose the position of seat that they want to buy. | | 2 | User click on the available seat with number of seat that they select in previous step. | System will select that seat and change it to green for user.  [Exception 1] |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | The actor cannot leave a seat empty between 2 selected seats. | System shows alert message: “Bạn không thể bỏ ghế ở giữa”. | | 2 | The actor does not select enough seat. | Sytem show alert message: “Bạn chưa chọn đủ ghế”. |   **Relationships:** Include with book ticket.  **Business Rules:**N/A | | | |
| Table 43: USE CASE - UC\_C004 <Customer> Choose Seat | | | |

* + - * 1. <Customer> Checkout (UC\_C005)

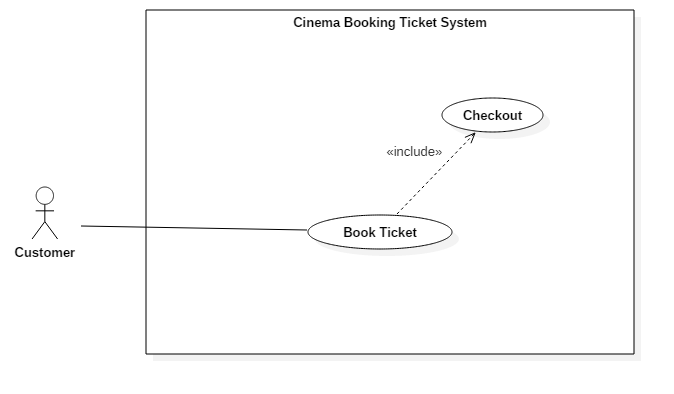


Figure 35: <Customer> Checkout (UC\_C005)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_C005** | | | |
| **Use Case No.** | UC\_C005 | **Use Case Version** | 2.0 |
| **Use Case Name** | Checkout | | |
| **Author** | DanhLC | | |
| **Date** | 20/06/2018 | **Priority** | High |
| **Actor:**   * Customer   **Summary:**   * This use case helps actor to check out the ticket that they want to buy.   **Goal:**   * Customer will own the ticket after this use case success.   **Triggers:**   * Actor sends check out command.   **Preconditions:**  **Post Conditions:**   * **Success:** The action is completed. * **Fail:** Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At choose seat screen. | System requires information from actor:   * **Phone**: free text input, required, length 9-11 digits. * **Email**: free text input, required, length 8-40 characters. | | 2 | Actor inputs information. |  | | 3 | Actor sends command to check out. | System validates information.  [Exception 1, 2] | |  | Confirm ticket dialog will open | System display confirm ticket information that user has choose. | | 4 | User click paypal button  [Alternative 1] | Paypal window will open and require user to specify their paypal’s account.  [Exception 3, 4] | | 5 |  | System create order and automatic send ticket code and order code for customer via email |   **Alternative Scenario:**  [alternative 1]   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | User click close button | Confirm dialog is closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Invalid Email format | System shows error message: “Email cannot be blank and must be 8-40 characters long and must match email format”. | | 2 | Invalid Phone format. | System shows error message: “Phone cannot be blank or contain alphabetical characters and must be 9-11 digits, ”. | | 3 | User has 8 minute for payment after confirming dialog are display. If time out the payment can’t complete | System shows error message: “Bạn đã hết thời gian đặt vé, vui lòng đặt vé lại”. |   **Relationships:** include with booking ticket  **Business Rules: N/A** | | | |
| Table 44: USE CASE - UC\_C005 <Customer> Checkout | | | |

* + - * 1. <Customer> Resell ticket (UC\_C006)

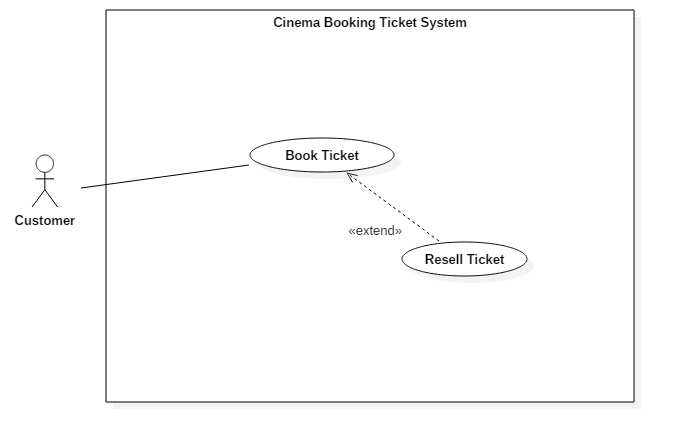


Figure 36: <Customer> Resell ticket (UC\_C006)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_C006** | | | |
| **Use Case No.** | UC\_C006 | **Use Case Version** | 2.0 |
| **Use Case Name** | Resell Ticket | | |
| **Author** | DanhLC | | |
| **Date** | 21/01/2016 | **Priority** | High |
| **Actor:**   * Staff   **Summary:**   * This use case can help customer and user can resell their ticket that they already buy. When they resell seat success. The seat will change color and information of buyer will show on that seat. The other customer wants to re-buy that seat can contact with them.   **Goal:**   * Customer can resell their ticket or the other customer.   **Triggers:**   * Customer send request to resell ticket.   **Preconditions:**  **Post Conditions:**   * **Success:** The action is completed. * **Fail:** Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At fixed header of here click “bán lại vé” button. | System requires enter email for see booking ticket history. | | 2 | Customer enter email to check. | System send confirm code for this email.  [Exception 1,2] | | 3 |  | System required enter confirm code. | | 3 | Customer enter code. | System validate confirm code and show all ticket belong to this email.  [Exception 3] | | 4 | Customer chooses the seat that want to resell. | System required customer enter the description for the ticket that customer intent to resell. | | 4 | Actor click “Bán lại vé”  [alternative 1] | System change seat status to “reselling” and change current seat color to blue. Other customer can see the blue seat and information of seller when they are booking ticket. | | 5 | After buyer and seller make agreement. Seller enter buyer email and click “xác nhận bán” | System change seat status to “reselled” and change current seat color to gray. |   **Alternative Scenario:**  [alternative 1]   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Actor click “Hoàn tất, chờ người mua liên hệ” | System validates ticket information.  [Exception 2] | |  |  | System automatic send new ticket code for the email that actor has inputted and cancel old ticket code. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Invalid email format | System shows error message: “email chưa đúng định dạng”. | | 2 | Invalid confirm code | System shows error message: “mã xác nhận bạn nhập chưa đúng”. |   **Relationships:** extend with booking ticket  **Business Rules:**   * System will auto generate new ticket code, send it for buyer, cancel old code. * Ticket will still belong to the old order. But customer can’t use resell ticket any more. | | | |
| Table 45: USE CASE - UC\_C006 <Customer> Resell ticket | | | |

* + - * 1. <Customer> Change ticket (UC\_C007)

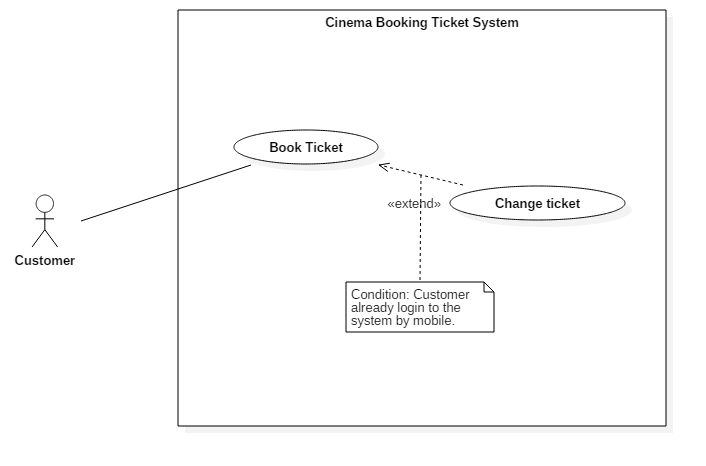


Figure 37: <Customer> Change ticket (UC\_C007)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_C007** | | | |
| **Use Case No.** | UC\_C007 | **Use Case Version** | 2.0 |
| **Use Case Name** | Change Ticket | | |
| **Author** | DanhLC | | |
| **Date** | 21/01/2016 | **Priority** | High |
| **Actor:**   * Staff   **Summary:**   * This use case can help customer and user can change schedule for this ticket before show time start 4 hours. The film they can change must be the same with the film they has bought.   **Goal:**   * Customer can change ticket for later schedule.   **Triggers:**   * Customer send request to change ticket.   **Preconditions:**  **Post Conditions:**   * **Success:** The action is completed. * **Fail:** Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At user history screen, user selects one ticket for change and then clicks change ticket. | System will show dialog with message “bạn có muốn đổi lại vé này không?” | | 2 | User chooses “đồng ý”.  [Alternative 1] | System show all schedule available for change ticket. | | 3 | User select one schedule for change | System goes to choose seat screen. | | 4 | User chooses one seat for ticket. |  | | 5 | User chooses next. |  | | 6 |  | System change seat for appropriate with your information that you has inputted. |   **Alternative Scenario: N/A**   |  |  |  | | --- | --- | --- | | Step | Cause | System Response | | 1 | Actor click “Hủy” | System close dialog. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Invalid ticket code | System shows error message: “Mã vẽ bạn vừa nhập sai!”. |   **Relationships:** include with booking ticket.  **Business Rules:**   * User can only change the ticket before the show time start 4 hours. * User can only change ticket once. * The schedule will be exchanged for ticket must belong to selected day. * User can only change ticket belong to current cinema. | | | |
| Table 46: USE CASE - UC\_C007 <Customer> Change ticket | | | |

* + - 1. <Cinema manager> Overview User Case

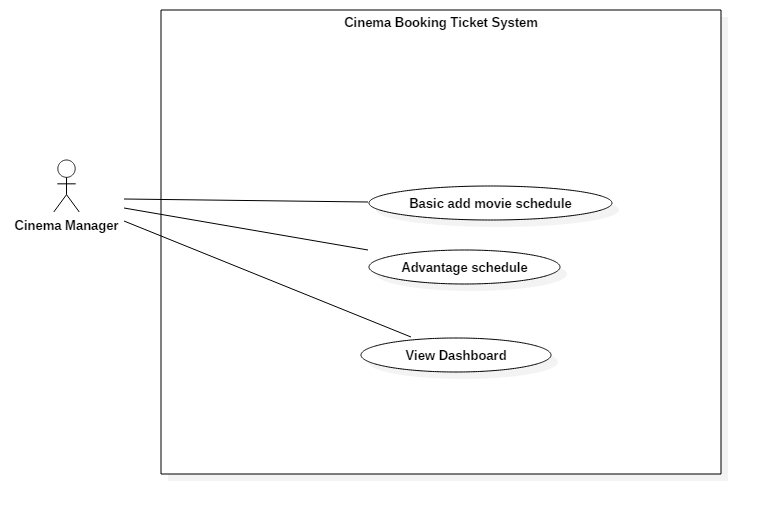


Figure 38: <Cinema Manager> Overview Use Case

* + - * 1. <Cinema manager > Basic add movie schedule (UC\_M002)

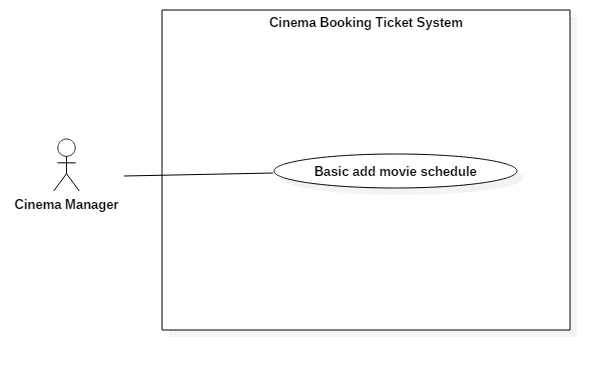


Figure 39: <Cinema manager > Basic add movie schedule (UC\_M002)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_M001** | | | |
| **Use Case No.** | UC\_M002 | **Use Case Version** | 2.0 |
| **Use Case Name** | Basic add movie schedule | | |
| **Author** | DanhLC | | |
| **Date** | 21/06/2018 | **Priority** | Normal |
| |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Actor:**   * Cinema manager   **Summary:**   * This use case helps actor to add movie schedule for their cinema. The schedule after that can update on customer view, so customer can book ticket for that schedule.   **Goal:**   * Actor can add schedule for their cinema.   **Triggers:**   * Actor sends sign in command.   **Preconditions:**   * Actor must login with cinema manager role.   **Post Conditions:**   * **Success:** Action is completed. * **Fail:** Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home page, Actor click “basic add schedule” button. | System show the screen for basic add schedule, and request following information:   * Select film from drop down list * Select show time from drop down list * Select room from drop down list * Select date from date time picker. | | 2 |  | System validates inputted information  [Exception 1] | | 3 | Actor click button “add schedule” | System add schedule with given information. |   **Alternative Scenario:**  **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | Schedule already existed. | System shows error message: “Schedule already existed.” |   **Relationships:** N/A  **Business Rules:**   * The Schedule will be added must not be duplicate or piled up to the other added schedule. | | | | |
| Table 47: USE CASE - UC\_M002 <Cinema manager > Basic add movie schedule | | | |

* + - * 1. <Cinema manager > Advantage schedule (UC\_M003)

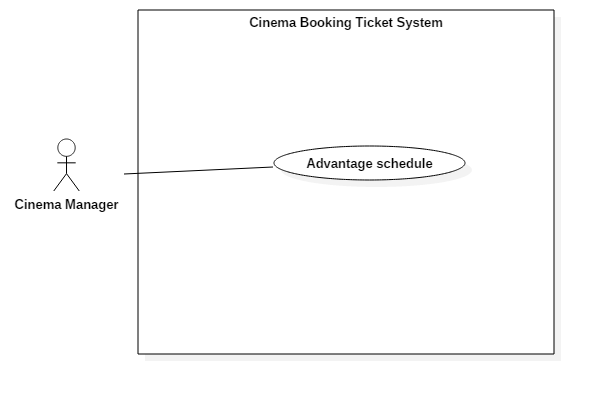


Figure 40: <Cinema manager > Advantage schedule (UC\_M003)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_M003** | | | |
| **Use Case No.** | UC\_M003 | **Use Case Version** | 2.0 |
| **Use Case Name** | Advantage schedule | | |
| **Author** | DanhLC | | |
| **Date** | 21/06/2018 | **Priority** | Normal |
| **Actor:**   * Cinema manager   **Summary:**   * This use case helps actor to add schedule for cinema in beauty and interactive interface. Besides that, by using beauty schedule algorithm the system will automatically calculate to suggest the best schedule for current selected day.   **Goal:**   * Actor can add schedule for current cinema.   **Triggers:**   * Actor sends add promotion command.   **Preconditions:**   * Actor must login with cinema manager role.   **Post Conditions:**   * **Success:** Action is completed. * **Fail:** Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home page, Actor click “advantage schedule” button | System request select date for suggest schedule. | | 2 | Actor a day | System will automatic show interactive interface for user to select film.  The purple cell is the suggesting film schedule  The blue cell is available schedule cell.  The red cell is already added schedule cell.  The green cell is choosing schedule. | | 3 | Actor click on blue cell | The blue cell will automatic change to green cell mean choosing, click again to unchosen  [Exception 1] | | 4 | Actor click “save” button  [alternative 1,2] | System will save all purple cell (suggested film) and green cell (choosing schedule) for current selected day to database. |   **Alternative Scenario: N/A**  *[alternative 1]*   |  |  |  | | --- | --- | --- | | Step. | Actor Action | System Response | | 1 | Actor choose “unsuggestion” button | System will change all purple cell to blue cell, mean that user no need system to suggest. |   *[alternative 2]*   |  |  |  | | --- | --- | --- | | Step. | Actor Action | System Response | | 1 | Actor choose “save for next day” button | System will save all purple cell (suggested film) and green cell (choosing schedule) for the next day of current selected day to database.  [exception 2] |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Cause | System Response | | 1 | No film has selected | System shows error message: “No film has been selected”. | | 2 | Next day already have schedule. | System shows error message: “Next day already have a schedule, system can’t override it”. |   **Relationships:** N/A  **Business Rules:**   * We can save schedule for next day if and only if next day has no schedule added before. * User select previous day from current day, system will automatic change to view mode. It’s mean we not provide any cell for user to click for adding schedule, we just show all schedule has been added before. * Once schedule has been added, we can’t edit or delete it. | | | |
|  | | | |
| Table 48: USE CASE - UC\_M003 <Cinema manager > Advantage schedule | | | |

* + - * 1. <Cinema manager > View dashboard (UC\_M004)

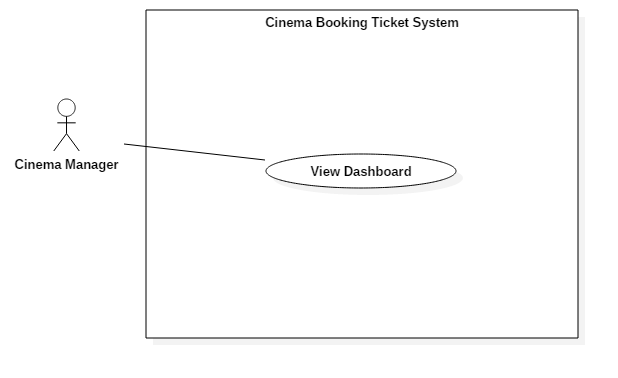


Figure 41: <Cinema manager > View dashboard (UC\_M004)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_M004** | | | |
| **Use Case No.** | UC\_M004 | **Use Case Version** | 2.0 |
| **Use Case Name** | View dashboard | | |
| **Author** | DanhLC | | |
| **Date** | 21/06/2018 | **Priority** | Normal |
| **Actor:**   * Cinema manager   **Summary:**   * This use case helps actor view dashboard. Dashboard will show basic information of this cinema. Dashboard will also statistic weekly sale ticket of current cinema and report top 4 film the system evaluate that is the best for this cinema.   **Goal:**   * Actor can view their cinema report.   **Triggers:**   * Actor sends view report command.   **Preconditions:**   * Actor must login with cinema manager role.   **Post Conditions:**   * **Success:** Action is completed. * **Fail:** Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | At home page, Actor click “view dashboard” button | System show flowing information   * Number of showing movie * Number of total room * Number of today schedule * Number of ticket sale today * Column chart for weekly ticket sale * Top 4 best film judge by the system. |   **Alternative Scenario: N/A**  **Relationships:** N/A  **Business Rules:** N/A | | | |
|  | | | |
| Table 49: USE CASE - UC\_M004 <Cinema manager > View dashboard | | | |

* + - 1. <System> Overview Use Case

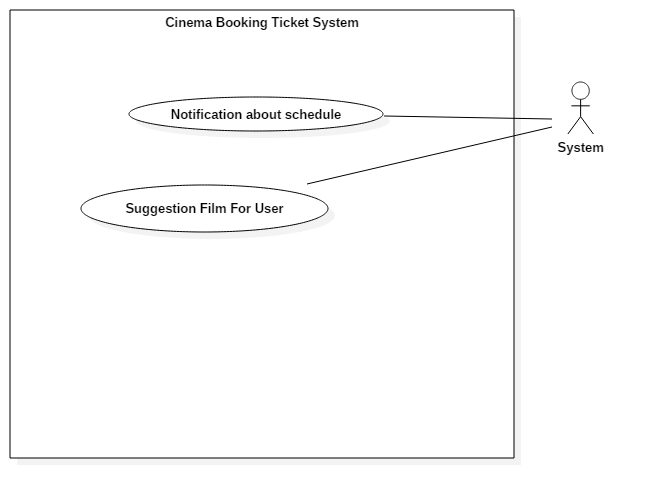


Figure 42: Initial <System> Overview Use Case

* + - * 1. <System> Notification about schedule (UC\_S002)

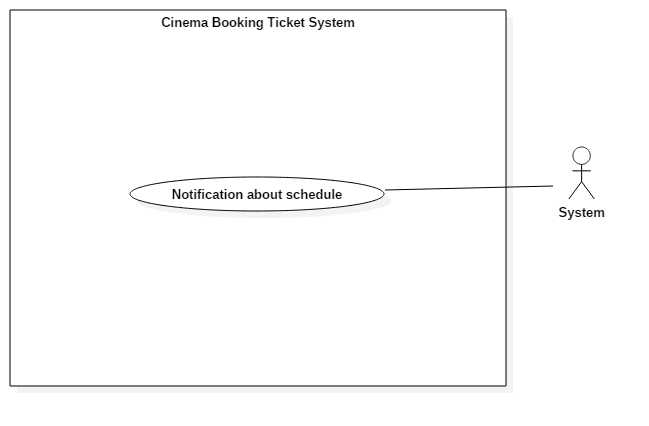


Figure 43: <System> Notification about schedule (UC\_S002)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_S002** | | | |
| **Use Case No.** | UC\_S002 | **Use Case Version** | 2.0 |
| **Use Case Name** | Notification about Schedule | | |
| **Author** | DanhLC | | |
| **Date** | 21/06/2018 | **Priority** | Normal |
| **Actor:**   * System   **Summary:**   * System will send automatic notification for mobile user before the film show time 30 minutes. The purpose is reminded them about the ticket that they have bought.   **Goal:**   * Notification for user about schedule.   **Triggers:**   * System runs success.   **Preconditions:**   * System deployed success.   **Post Conditions:**   * **Success:** Action complete. * **Fail:** Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | System deploys | System runs initial function notification for mobile user. |   **Alternative Scenario: N/A**  **Exceptions: N/A**  **Relationships:** N/A  **Business Rules:** N/A. | | | |
| Table 50: USE CASE - UC\_S002 <System> Notification about schedule | | | |

* + - * 1. <System> suggest film for user (UC\_S003)

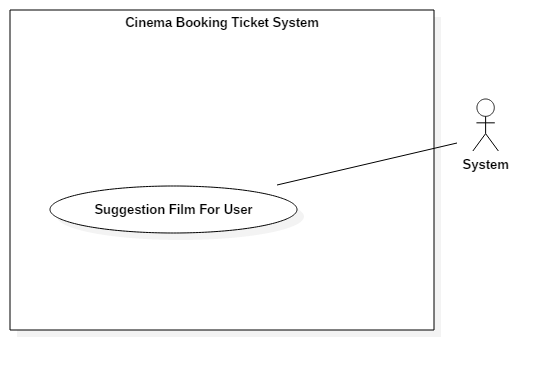


Figure 44: <System> suggest film for user (UC\_S003)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UC\_S003** | | | |
| **Use Case No.** | UC\_S003 | **Use Case Version** | 2.0 |
| **Use Case Name** | Suggest film for user | | |
| **Author** | DanhLC | | |
| **Date** | 21/06/2018 | **Priority** | Normal |
| **Actor:**   * System   **Summary:**   * System will automatic suggest film for user. When they go to home page list 4 films suitable with their favorite will show on top. Each week we will send mail to suggest film for them.   **Goal:**   * System automatic suggest film for user base on their favorite.   **Triggers:**   * System runs success.   **Preconditions:**   * System deploys success.   **Post Conditions:**   * **Success:** Action complete. * **Fail:** Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | System deploy | System run initial function for suggest film for user |   **Alternative Scenario: N/A.**  **Exceptions: N/A.**  **Relationships:** N/A.  **Business Rule: N/A.** | | | |
| Table 51: USE CASE - UC\_S003 <System> suggest film for user | | | |

* 1. Software Requirement Specification
     1. Usability
* Admin, Partner, cinema manager interface is written in English including all dialogs and messages.
* User interface are written in Vietnamese including all dialogs and messages.
* The system needs no more than 2 days of training for cinema manager.
  + 1. Reliability
* The system can allow ticket buyers to resell or return their ticket 4 hour prior to the movie show time in case they cannot go to the cinema.
* All account password will be encode in SHA-1 which takes an input and produces a 160-bit (20-byte) hash value known as a message digest - typically rendered as a hexadecimal number, 40 digits long.
  + 1. Availability
* Server has back-up method to protect and restore necessary data as soon as possible whenever there is a problem.
  + 1. Security.
* Each role of user has a specific permission to interact with the system.
* System always checks for authorization and authentication before doing anything.
* Only Admin can create partner account, only partner can create cinema account.
  + 1. Maintainability
* The system is divided into separated modules.
  + 1. Portability
* User can use the mobile application on devices running Android 6 or later.
* Web application can be run on Chrome browser version 42 or later.
  + 1. Performance
* Client System runs fast and returns results in less than 2 seconds under 4Mbps bandwidth.
* Admin System run fast and returns results in less than 2 seconds under 4Mbps bandwidth.
  1. Conceptual Diagram

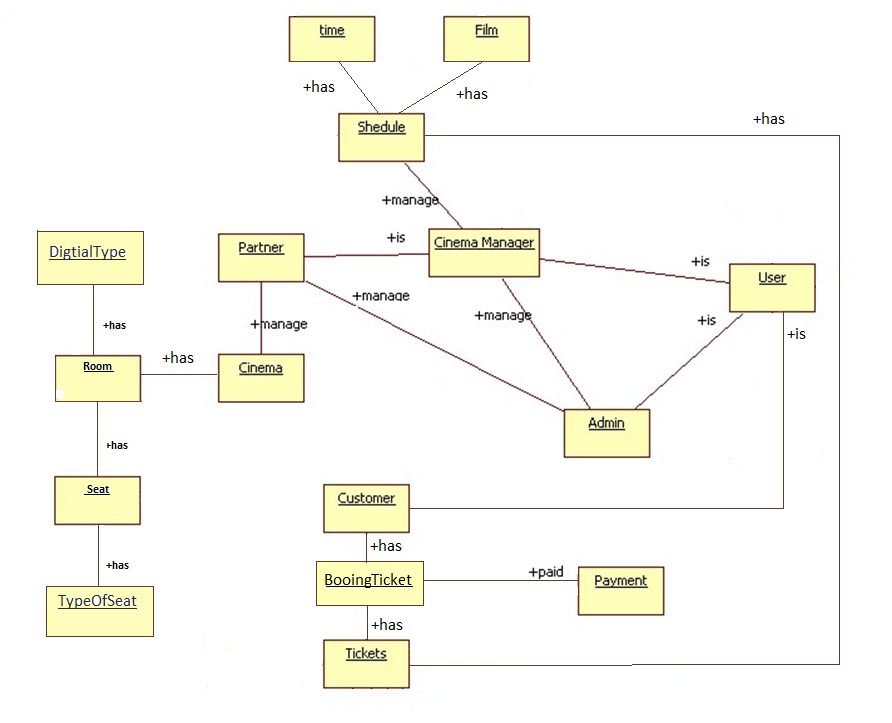


Figure 45: Conceptual Diagram

|  |  |  |
| --- | --- | --- |
| **Entity Name** | **Description** | |
| Admin | Person who manages the system. | |
| Cinema Manager | Person who manages the schedule of the cinema. | |
| Partner | Person who managers the cinema in their system | |
| User | Person who manages their profile and see their booking history | |
| Customer | Person who books tickets. | |
| Payment | Contains all payment methods. | |
| Seat | Contains all information’s of seat. | |
| Cinema | Contains all information’s of cinema. | |
| Film | Contains all information’s of film. | |
| Tickets | Contains information that customer booking about cinema, film, seat and time | |
| Room | Contains all information’s of room | |
| Schedule | Contains all information’s schedule | |
| time | Contains all information of show time. | |
| Type Of Seat | Contains all information’s of seat type | |
| Digital Type | Contains all information’s of digital type | |
| Booking Ticket | Contains all information’s of order | |
| Table 52: Conceptual Diagram Data Dictionary | |

1. Software Design Description
   1. Design Overview

* The architectural design describes the overall architecture of the system and the architecture of each main component and subsystem.
* The detailed design describes static and dynamic structure for each component and functions. It includes class diagrams, class explanations and sequence diagrams for each use cases.
* The database design describes the relationships between entities and details of each entity.
* Document overview:
* Section 1: gives an overall description of the system architecture design.
* Section 2: gives component diagrams that describe the connection and integration of the system.
* Section 3: gives the detail design description, which includes class diagram, class explanation, and sequence diagram to details the application functions.
* Section 4: gives the interface design description, which includes component interface, web application interface and mobile application design.
* Section 5: describe a fully attributed Entity Relationship Diagram.
* Section 6: describe the algorithms that apply to the system.
  1. System Architectural Design
     1. Web Application Architecture Description

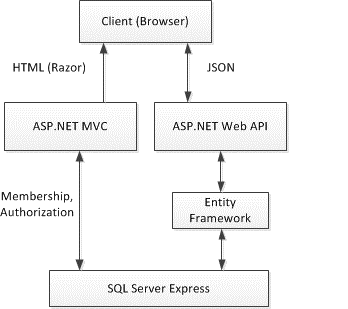


Figure 46: Web Application Architecture

We choose this architecture for a couple of reasons:

* To maximize the render performance of web client. Our application may need most of the time to do training
* Easy to develop, this architecture separates frontend out of backend. It helps developers can work independently with each other
* The architecture divides the application into components. This increases reusability and testability
* Reduce side effects in application
  + 1. Web Service Application Architecture Description

Backend Architecture

*Figure 47: Web Service Application Architecture*

The Web Service Application Architecture will base on MVC architecture. We choose this architect because of following advantages:

* The application will be divided into three clear part
* Testing each component in the architecture become easier
* The existing of Model or Business Layer leverage testability and decoupling to the View
* Active Record is a pattern to abstract the data layer and manipulate the database; it also acts as an ORM. It combines both data and domain logic into object so that user can easily know how to update the data
  + 1. Mobile Application Architecture Description

Mobile Architecture

*Figure 48: Mobile Application Architecture*

Mobile application architecture will use MVVM. There is couple of reasons choosing this architect:

* It minimizes the number of dependency, helps increase loose coupling between application
* The View is passive and will be notified to update data
* The View Model will contain complex view logic; it also has no information about View. This help increase reusability of View Model
* The Model is where business logic sit in; it also performs API requests to backend application

Develop Mobile Application using MVVM have increase productivity and reduce side effects

* 1. Component Diagram

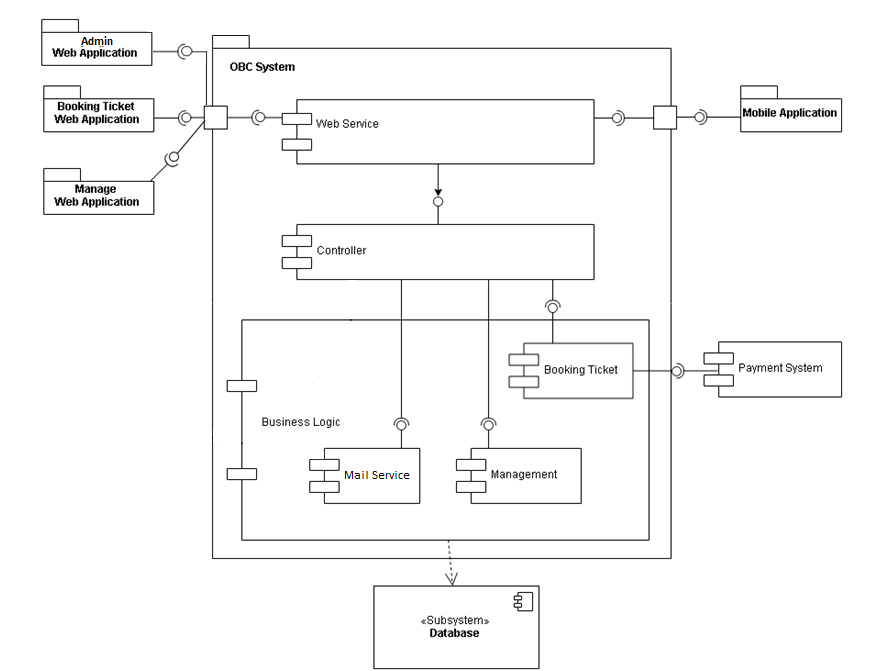


Figure 49: Component Diagram

|  |  |
| --- | --- |
| **Name** | **Description** |
| OBS System | Online cinema booking system |
| Booking Ticket Web Application | Package application to use system |
| Manage Web Application | Package application to manage system |
| Admin Web Application | Package application to admin system |
| Mobile Application | Package application for mobile device to use system |
| Web Service | Component is used to provide API for mobile application to interact with the system. |
| Business Logic | Common services are used to handle system’s business operations |
| Controller | Component to handle HTTP request |
| Management | Component to manage system data |
| Booking Ticket | Component to choosing ticket |
| Payment System | External component for handle purchasing |
| Mail System | External component for message to send email |
| Database System | Where application data is manipulated and stored |
| Table 53: Component Dictionary | |

* 1. Detailed Description

1. 4. 1. Class Diagram

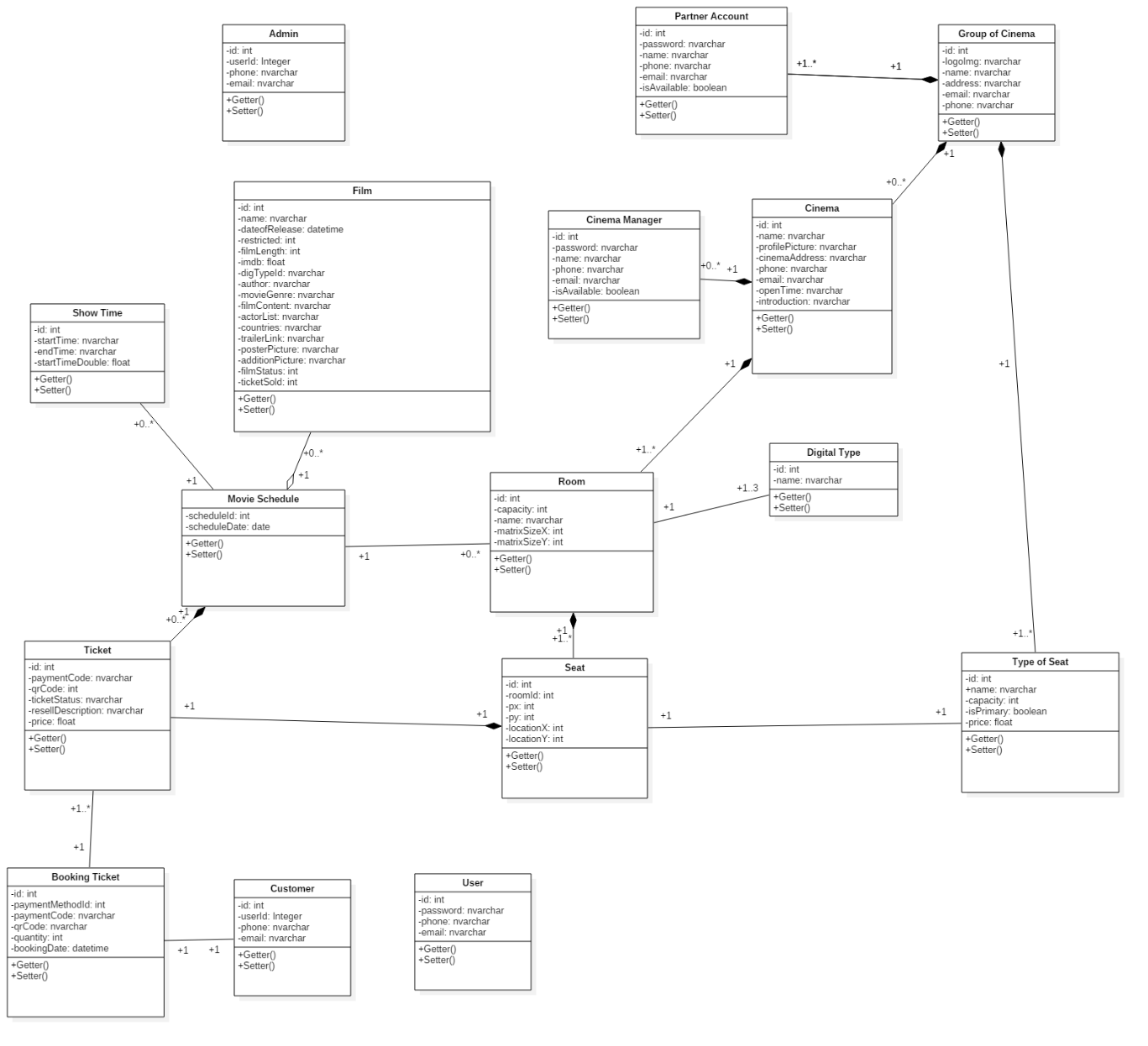


Figure 50: Class Diagram

* + 1. Class Diagram Explanation

|  |  |  |
| --- | --- | --- |
| **Class name** | **Mapping column with conceptual diagram** | **Description** |
| User Entity | User Account | Contains the information of users in the system. |
| Admin Entity | Admin Account | Contains the information of admin in the system. |
| Customer Entity | Customer | Contains the information of customers in the system. |
| Partner Entity | Partner Account | Contains the information of partners in the system. |
| Cinema Manager  Entity | Cinema Manager | Contains the information of cinema managers in the system. |
| Film Entity | Film | Contains the detailed information of all films in the systems. |
| Movie schedule Entity | Movie schedule | Contains the information of all movie schedules in the system. |
| Show Time Entity | Show Time | Contains the detailed information of all film show time in the systems. |
| Room Entity | Room | Contains the detailed information of all room in the systems. |
| Digital Type Entity | Digital Type | Contains the information of all the type of digital in the systems. |
| Seat Entity | Seat | Contains the information of all seats in the systems. |
| Type of Seat Entity | Type of Seat | Contains the information of all the types of seat in the systems. |
| Group Cinema Entity | Group Cinema | Contains the information of all the groups of cinema in the systems. |
| Cinema Entity | Cinema | Contains the information of all the cinemas in the systems. |
| Ticket Entity | Ticket | Contains the information of all the tickets in the systems. |
| Booking Ticket Entity | Booking Ticket | Contains the information of all the transactions when booking ticket. |
| Table 54: Class Diagram Dictionary | | |

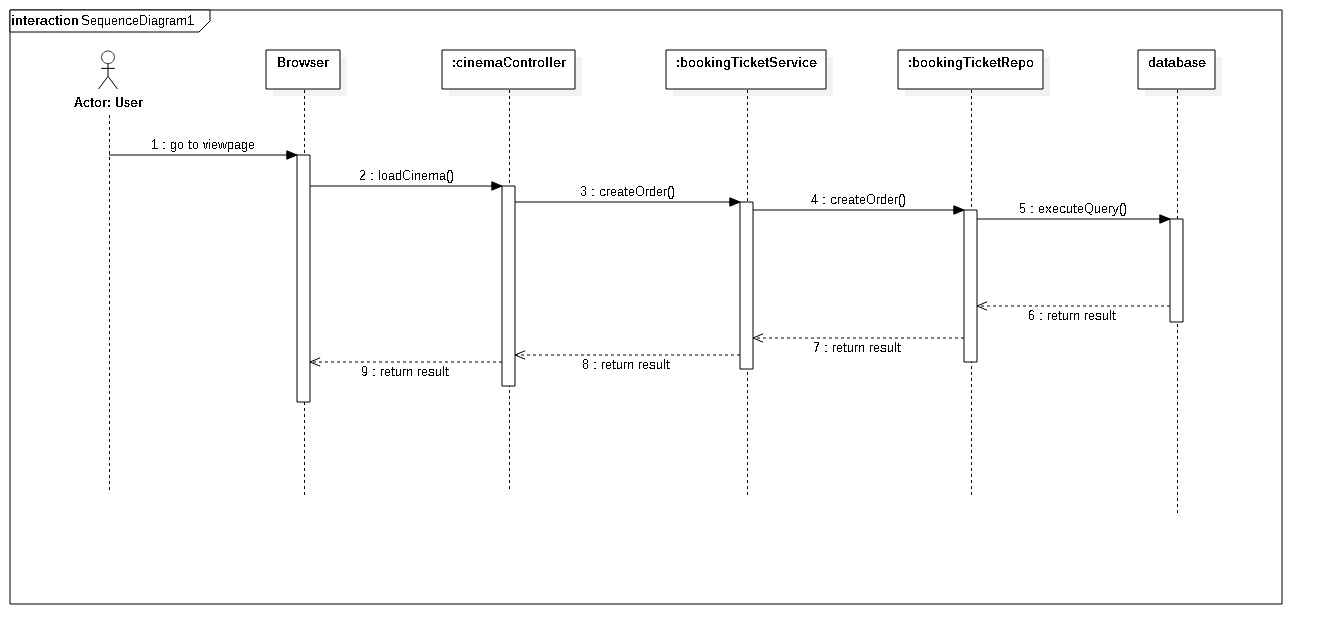
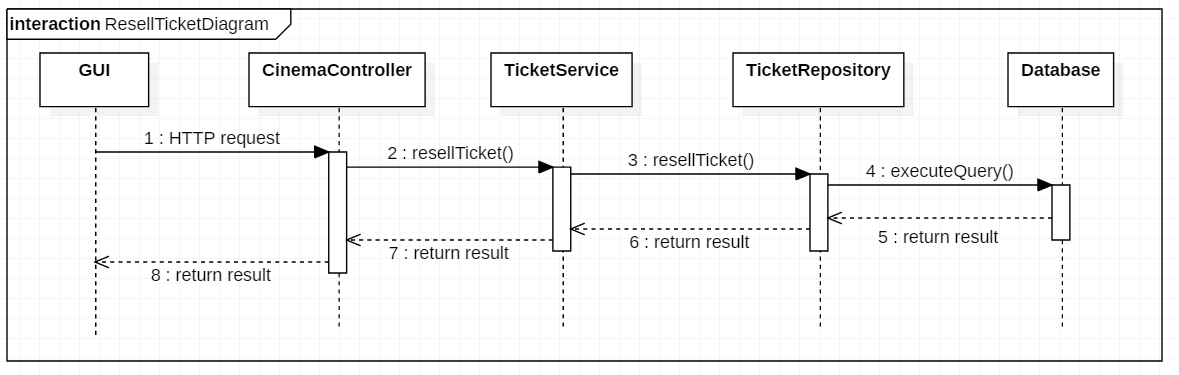
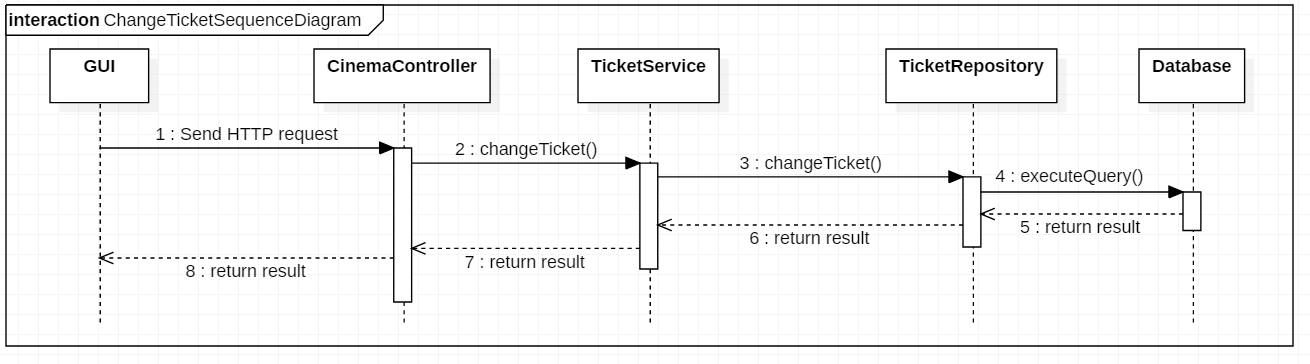
* + 1. Interaction Diagram
       1. Sequence Diagram
          1. <User> Booking ticket

Figure 51: Sequence Diagram - <User> Booking ticket

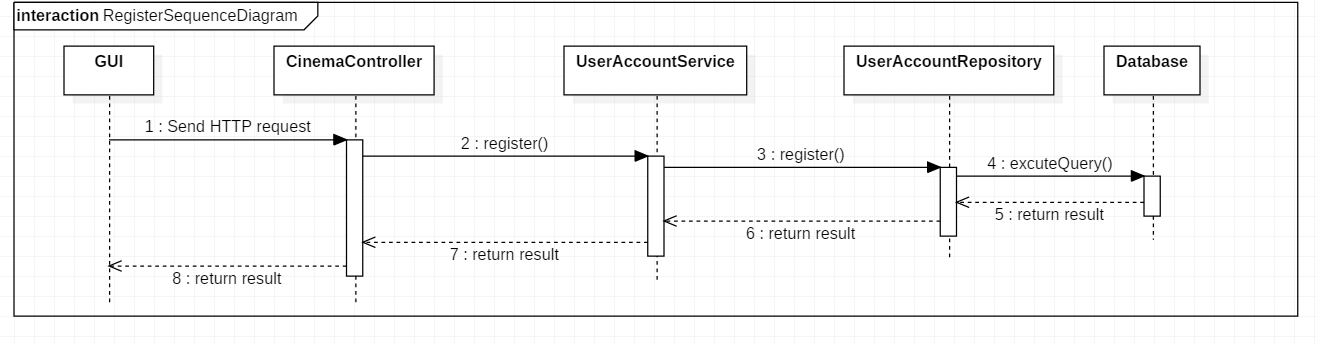
* + - * 1. <User> Resell ticket

Figure 52: Sequence Diagram - <User> Resell ticket

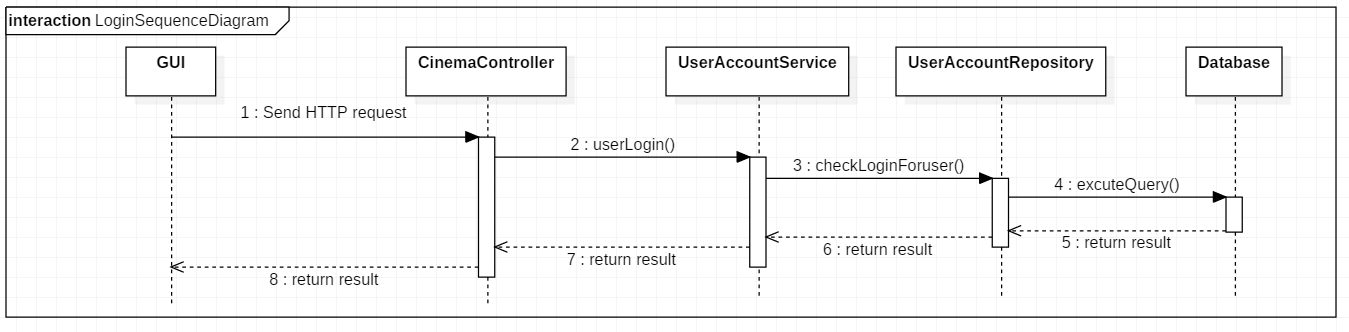
* + - * 1. <User> Change ticket

Figure 53: Sequence Diagram - <User> Change ticket

* + - * 1. <User> User register

Figure 54: Sequence Diagram - <User> User register

* + - * 1. <User> User login

Figure 55: Sequence Diagram - <User> User login

* + - * 1. <User> User logout

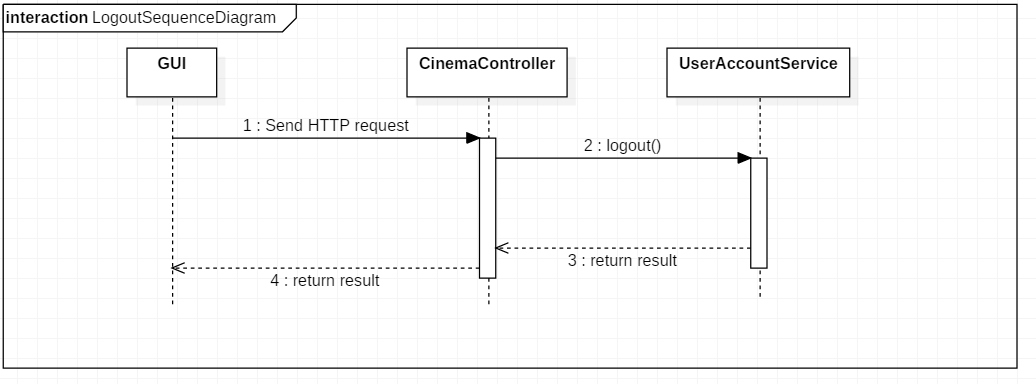


Figure 56: Sequence Diagram - <User> User logout

* + - * 1. <User> Edit profile



Figure 57: Sequence Diagram - <User> Edit profile

* + - * 1. <User> View booking ticket history

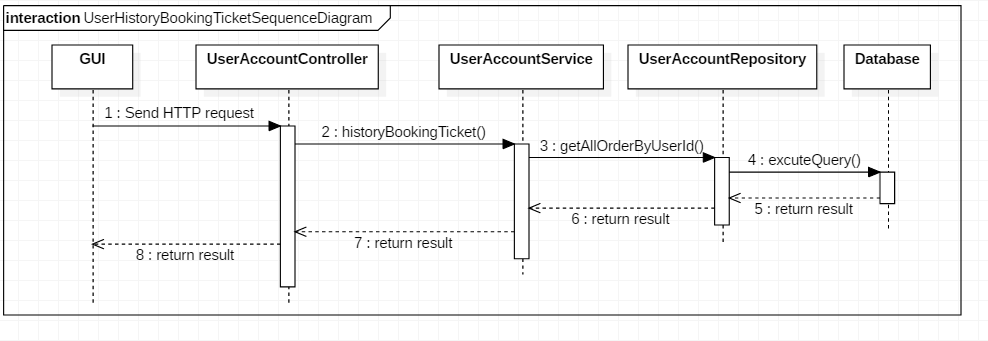


Figure 58: Sequence Diagram - <User> View booking ticket history

* + - * 1. <Partner> Partner login

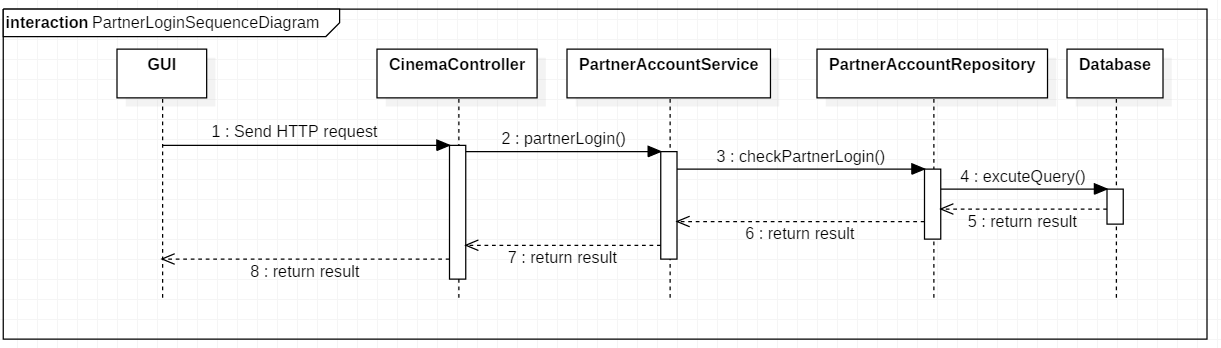


Figure 59: Sequence Diagram - <Partner> Partner Login

* + - * 1. <Partner> Partner create new cinema

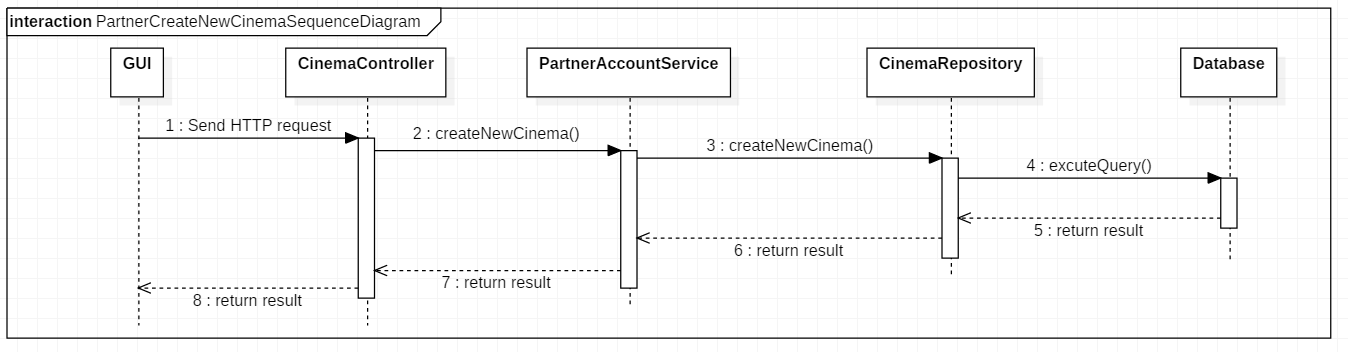


Figure 60: Sequence Diagram - <Partner> Partner create new cinema

* + - * 1. <Partner> Partner add room for cinema

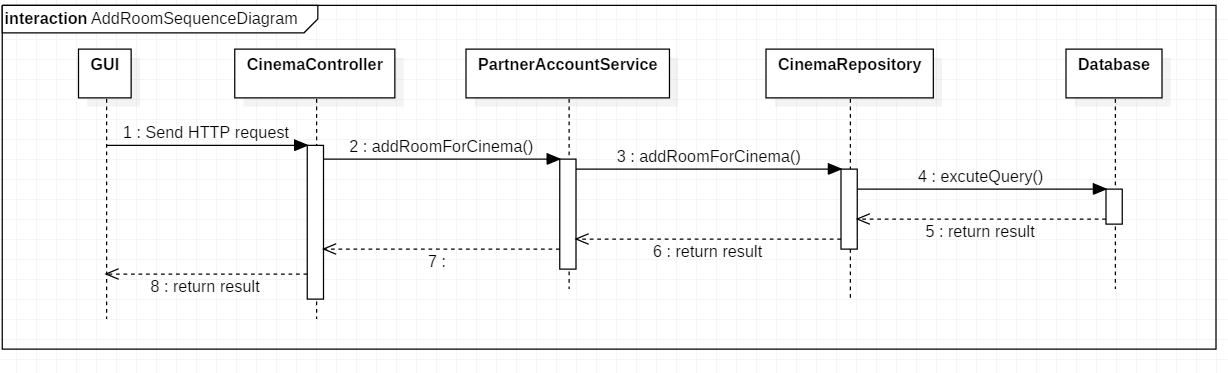


Figure 61: Sequence Diagram - <Partner> Partner add room for cinema

* + - * 1. <Partner> Partner edit cinema information

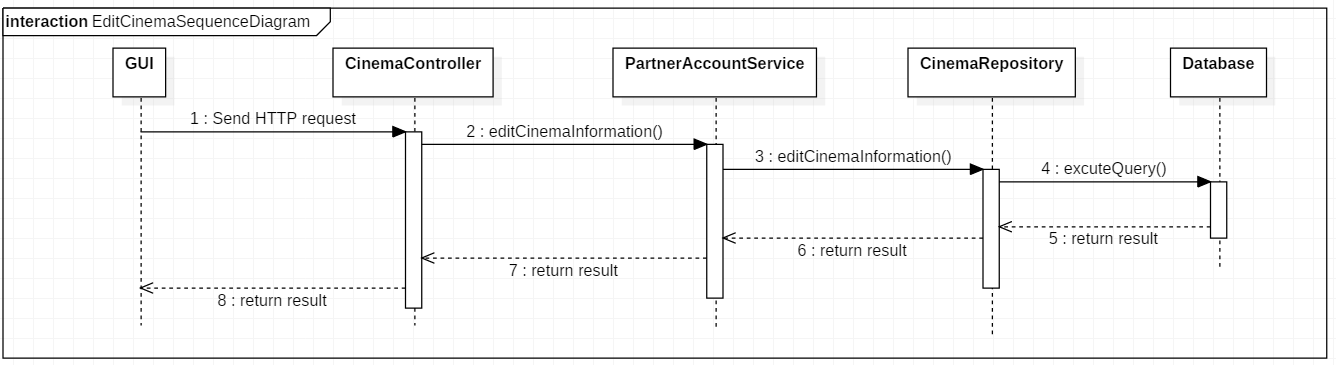


Figure 62: Sequence Diagram - <Partner> Partner edit cinema information

* + - * 1. <Partner> Partner add cinema manager account

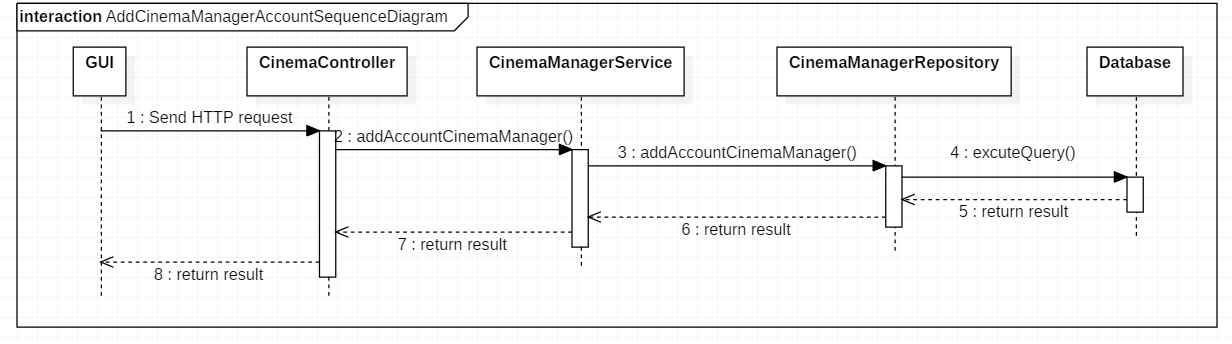


Figure 63: Sequence Diagram - <Partner> Partner add cinema manager account

* + - * 1. <Partner> Partner edit cinema manager account information

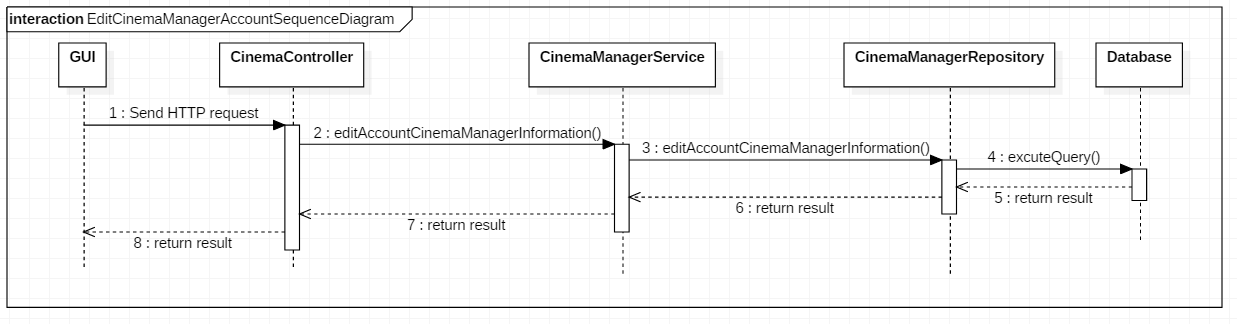


Figure 64: Sequence Diagram - <Partner> Partner edit cinema manager account information

* + - * 1. <Partner> Partner delete cinema manager account



Figure 65: Sequence Diagram - <Partner> Partner delete cinema manager account

* + - * 1. <Partner> Partner view report

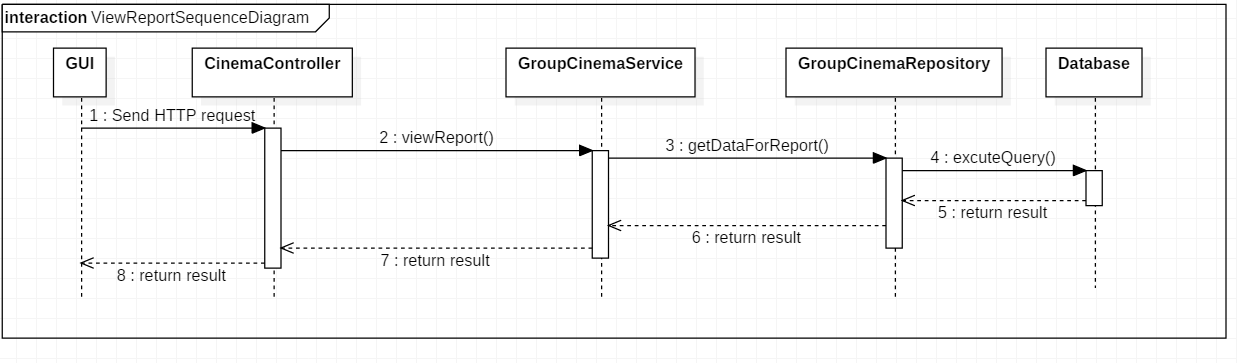


Figure 66: Sequence Diagram - <Partner> Partner view report

* + - * 1. <Admin> Admin login

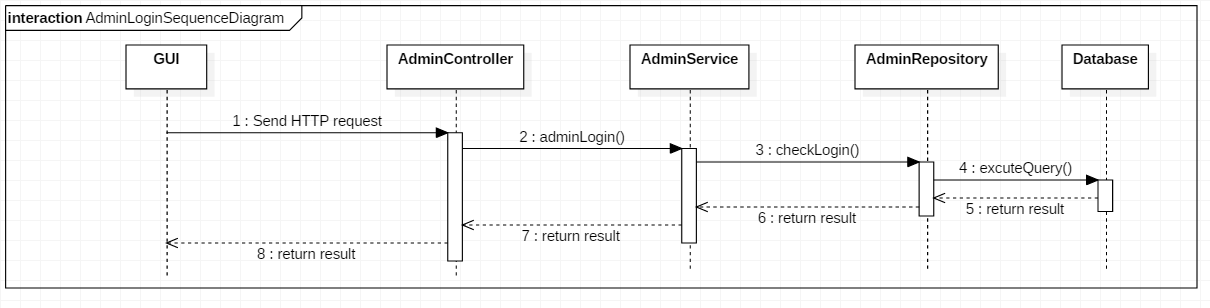


Figure 67: Sequence Diagram - <Admin> Admin login

* + - * 1. < Admin > Admin add film



Figure 68: Sequence Diagram - <Admin> Admin add a film

* + - * 1. < Admin > Admin edit film information

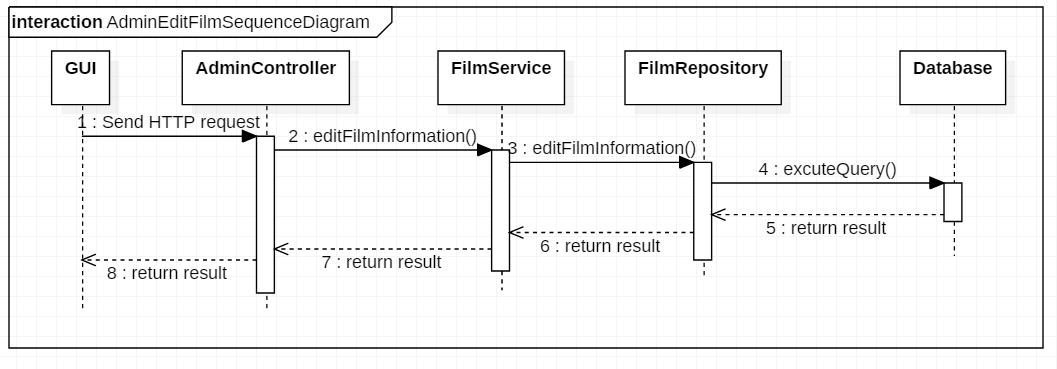


Figure 69: Sequence Diagram - <Admin> Admin edit film information

* + - * 1. < Admin > Admin add group cinema

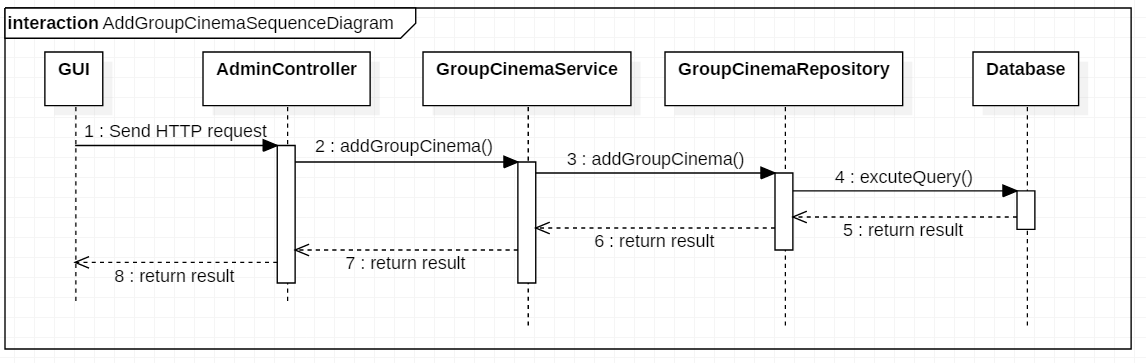


Figure 70: Sequence Diagram - <Admin> Admin add group cinema

* + - * 1. < Admin > Admin edit group cinema information

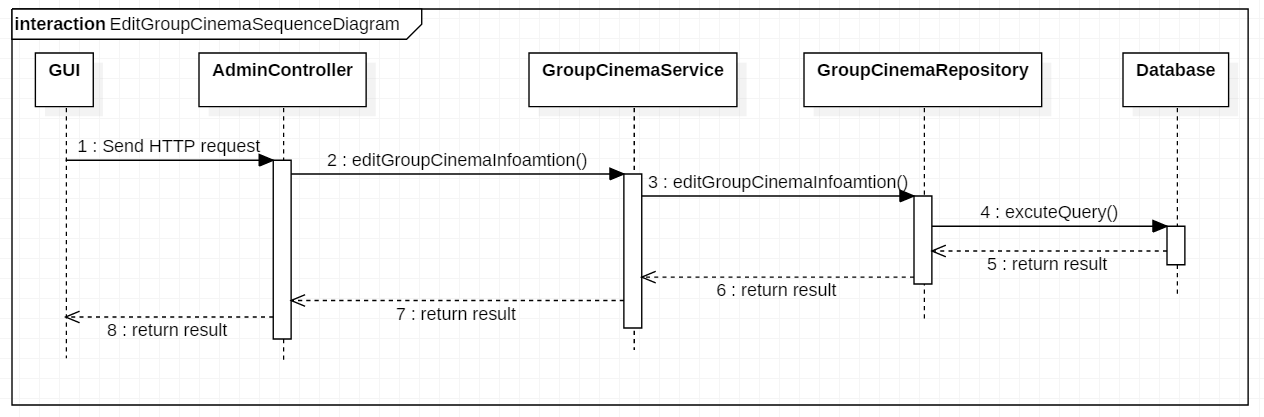


Figure 71: Sequence Diagram - <Admin> Admin edit group cinema information

* + - * 1. < Admin > Admin add partner account

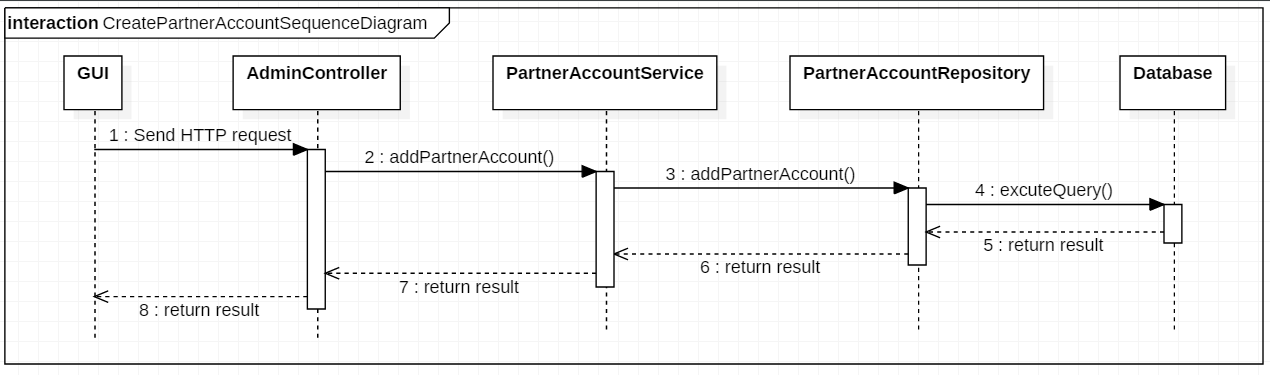


Figure 72: Sequence Diagram - <Admin> Admin add partner account

* + - * 1. < Admin > Admin edit partner account information

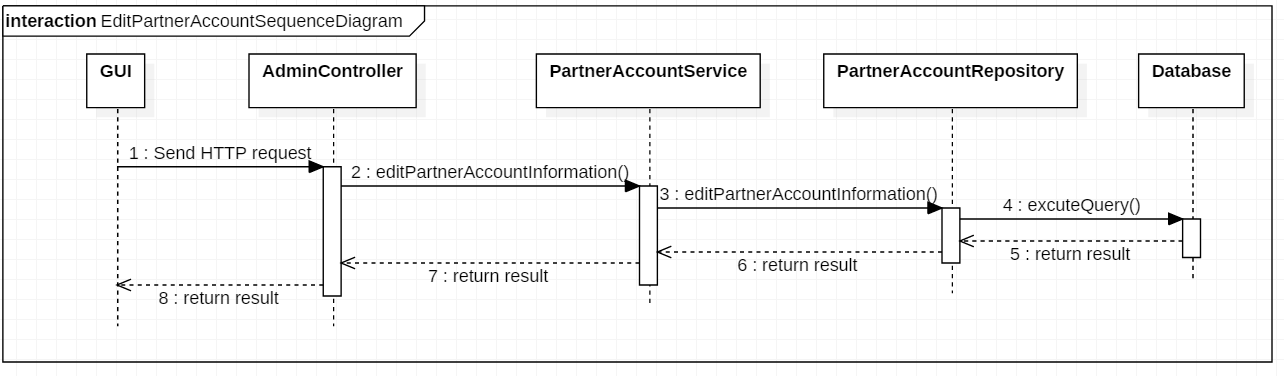


Figure 73: Sequence Diagram - <Admin> Admin edit partner account information

* + - * 1. < Admin > Admin delete partner account

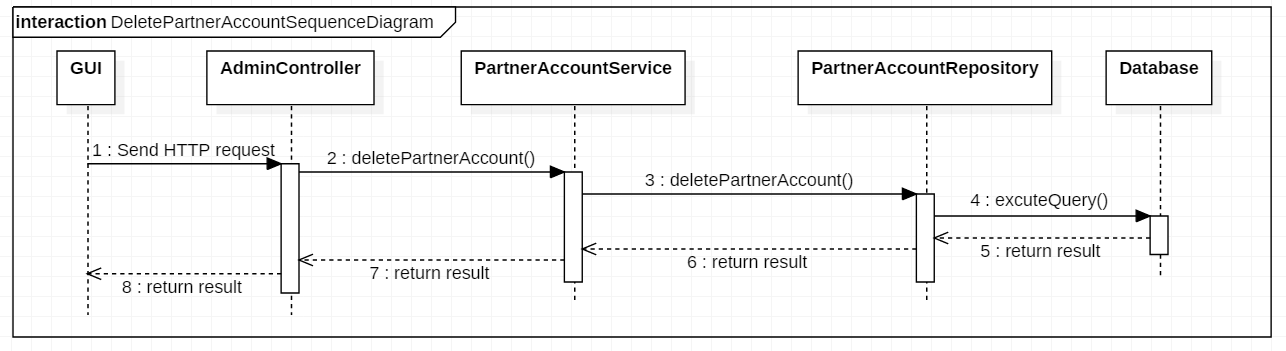


Figure 74: Sequence Diagram - <Admin> Admin delete partner account

* + - 1. Activity Diagram
         1. Login

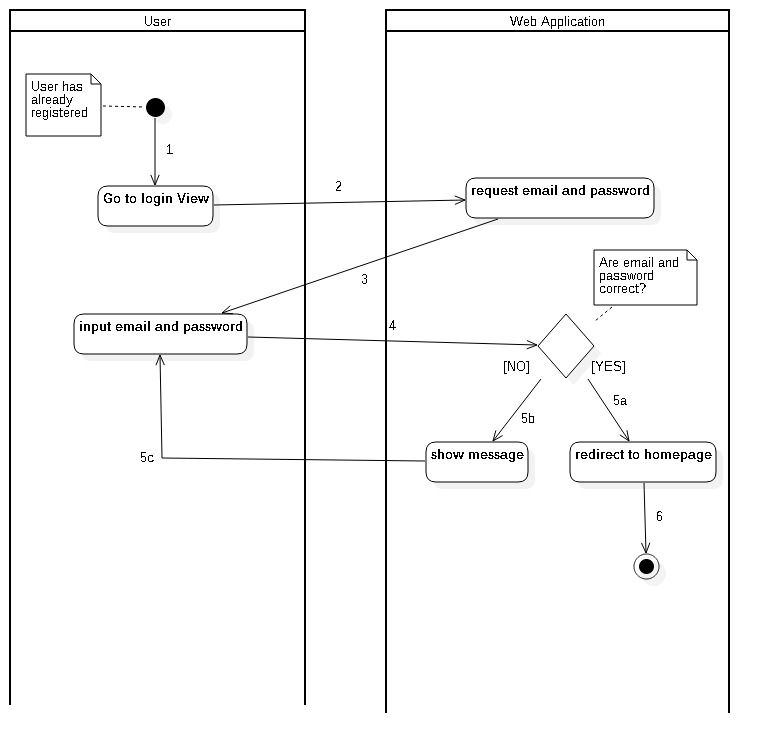


Figure 75: Activity Diagram - Login

* + - * 1. Booking Ticket Web Application

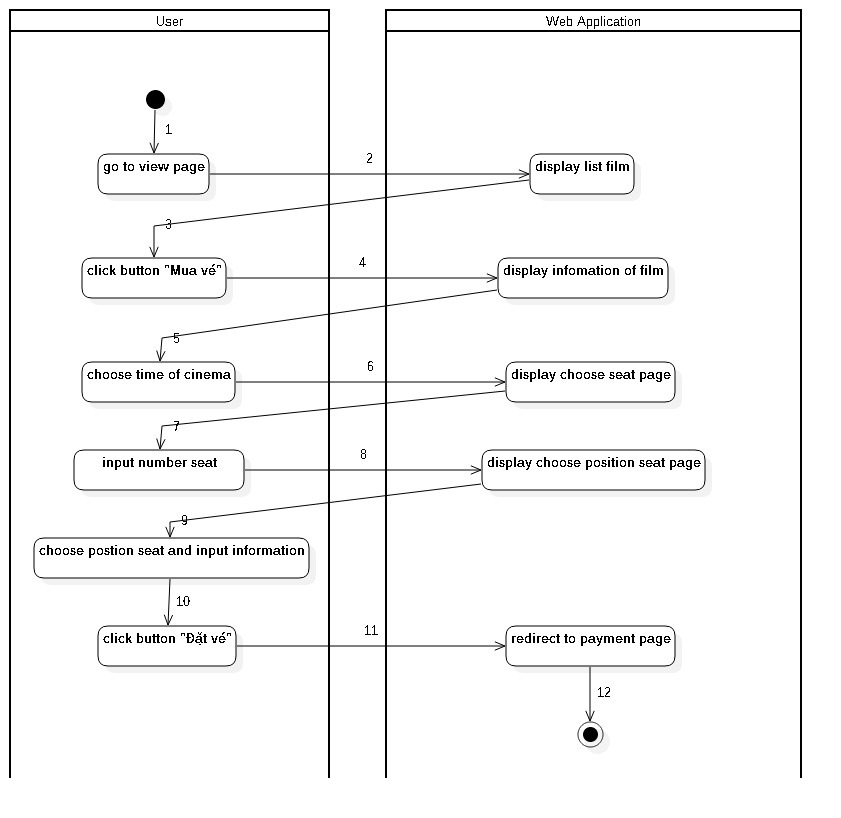


Figure 76: Activity Diagram – Booking Ticket Web Application

* + - * 1. Booking Ticket Mobile Application

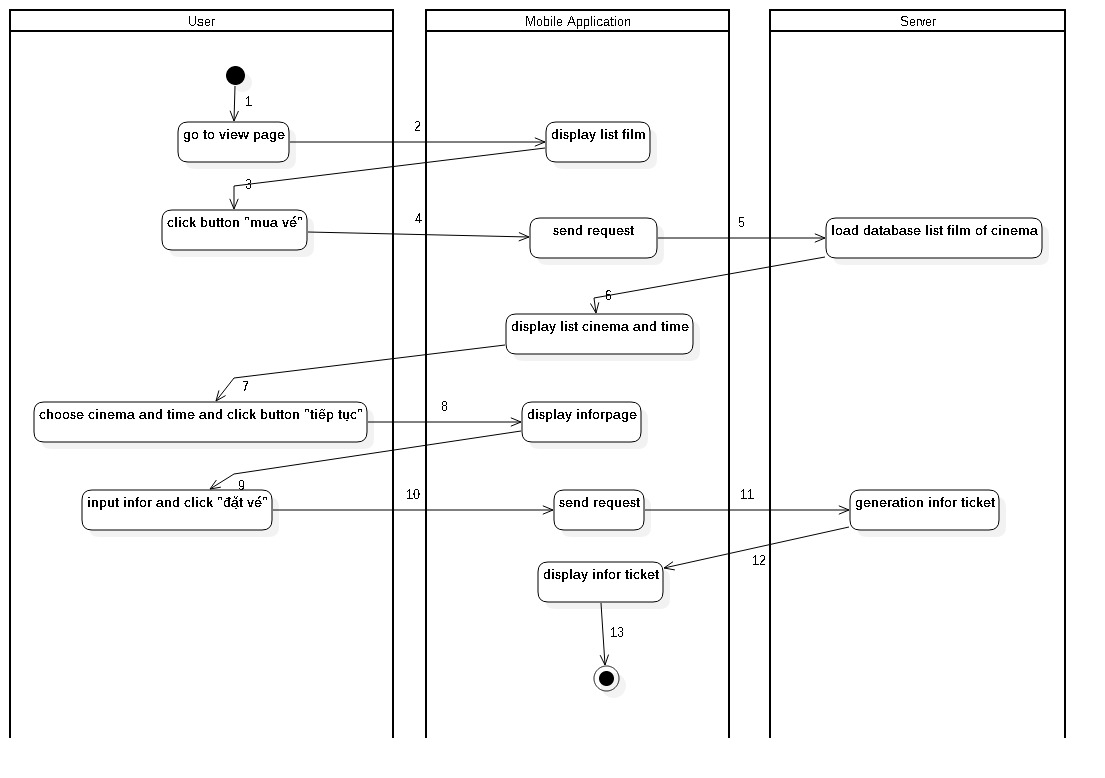


Figure 77: Activity Diagram – Booking Ticket Mobile Application

* 1. Interface
     1. Web Application
        1. User
           1. Home Page

|  |
| --- |
| screencapture-cinemabookingticket-azurewebsites-net-2018-08-08-09_01_51.png |

* + - * 1. Film Detail

|  |
| --- |
| screencapture-cinemabookingticket-azurewebsites-net-home-FilmDetail-2018-08-08-09_05_36.png |

* + - * 1. Choose Ticket

|  |
| --- |
| screencapture-cinemabookingticket-azurewebsites-net-home-ChooseTicketAndSeat-2018-08-08-09_15_22 (1).png |

* + - * 1. Choose Seat

|  |
| --- |
| screencapture-cinemabookingticket-azurewebsites-net-home-ChooseTicketAndSeat-2018-08-08-09_17_18.png |

* + - 1. Admin
         1. Manage Film

|  |
| --- |
| screencapture-admincinema-azurewebsites-net-2018-08-08-09_44_57.png |

* + - * 1. Manage Partner

|  |
| --- |
| screencapture-admincinema-azurewebsites-net-Home-ManagerPartnerAccount-2018-08-08-09_48_20.png |

* + - * 1. Manage Group of Cinema

|  |
| --- |
| screencapture-admincinema-azurewebsites-net-Home-ManagerGroupCinema-2018-08-08-09_49_39.png |

* + - 1. Partner
         1. Manage Cinema

|  |
| --- |
| screencapture-cinemamanagementapp-azurewebsites-net-2018-08-08-10_00_15.png |

* + - * 1. Manage Employee

|  |
| --- |
| screencapture-cinemamanagementapp-azurewebsites-net-2018-08-08-10_00_40.png |

* + - 1. Cinema Manager
         1. Manage Schedule

|  |
| --- |
| screencapture-cinemamanagementapp-azurewebsites-net-2018-08-08-10_04_21.png |

* + 1. Mobile Application
       1. Front Page

|  |  |
| --- | --- |
| Home Page | 1. Screen Home_1.PNG |
| Current Films | 2. Screen Đang Chiếu_1.PNG |
| Up-coming Films | 3. Screen Sắp Chiếu_1.PNG |
| Table : Mobile Application – Front Page - Interface | |

* + - 1. Cinema Page

|  |  |
| --- | --- |
| List of Group Cinema | 4. Screen Cụm rạp_1.PNG |
| List of Films by Cinema | 5. Screen Lịch chiếu theo rạp.PNG |
| Cinema Information | 6. Screen Thông tin rạp.PNG |
| Table : Mobile Application – Cinema Page - Interface | |

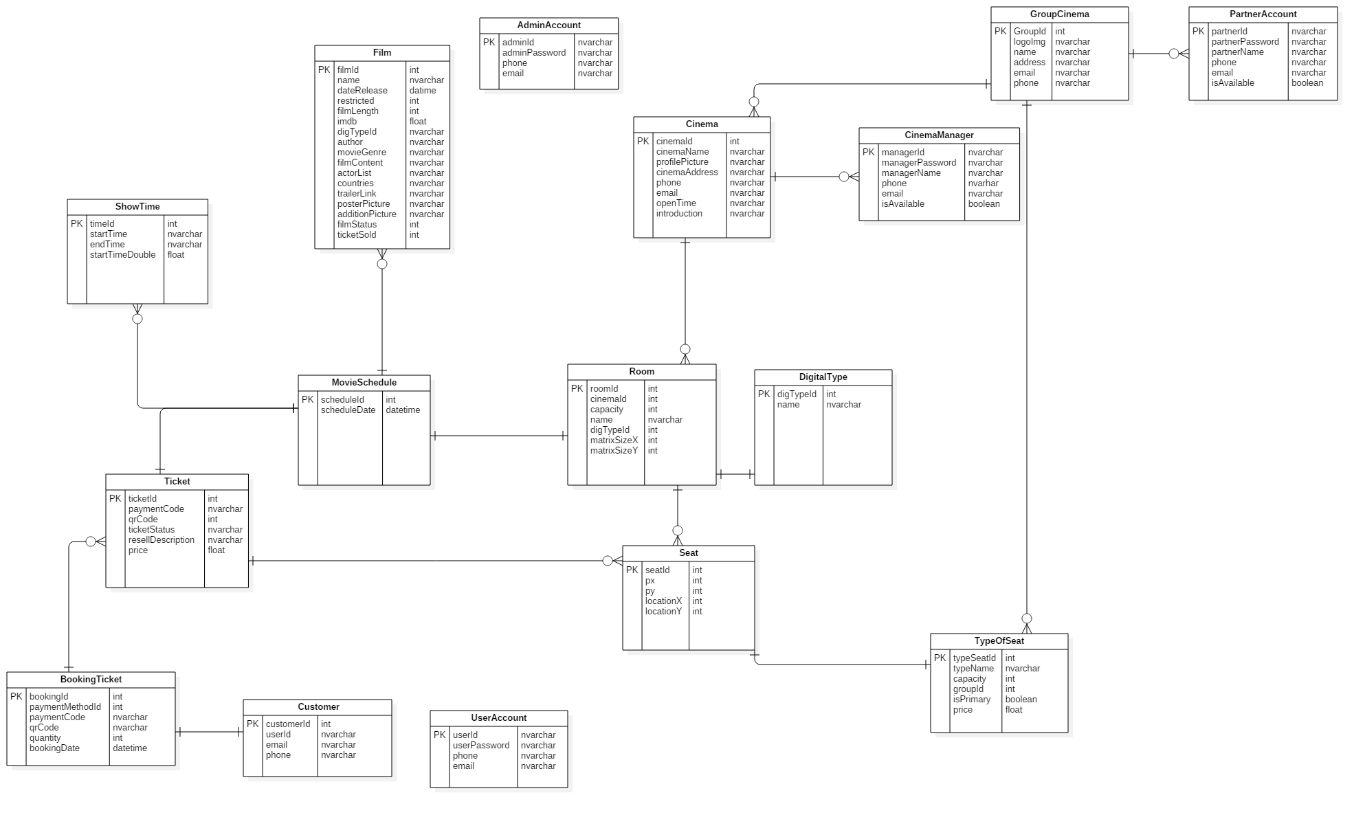
* + - 1. User Page

|  |  |
| --- | --- |
| Login | 11. Screen Đăng Nhập.PNG |
| Register | 18. Screen Đăng Kí.PNG |
| User Info | 8. Screen Cá Nhân (Đã Login).PNG |
| Purchased Tickets | 9. Screen Vé Đã Mua.PNG |
| Purchased Ticket Info | 10. Screen Chi tiết đơn hàng.PNG |
| Table : Mobile Application – User Page - Interface | |

* + - 1. Purchase Ticket Page

|  |  |
| --- | --- |
| Film’s Schedule | 12. Screen Lịch Chiếu Theo Film.PNG |
| Choose number of seats | 13. Screen Chọn số lượng ghế (Đặt vé).PNG |
| Choose the position(s) of seat(s) | 14. Screen Chọn ghế (Đặt vé).PNG |
| Confirmation | 15. Screen Xác nhận đơn hàng (Đặt vé).PNG |
| Choose type of payment | 16. Screen PayPal Payment 1 (Đặt vé).PNG |
| Purchased Ticket Succeeded | 17. Screen Đặt vé thành công (Đặt vé).PNG |
| Table : Mobile Application – Purchase Ticket Page - Interface | |

* 1. Database Design
     1. Entity Relationship Diagram (ERD)



*Figure 78: Entity Relationship Diagram (ERD)*

* + 1. Data Dictionary

|  |  |
| --- | --- |
| **Entity Data Dictionary: describe contents of all entities** | |
| **Entity Name** | **Description** |
| Film | Contains information of all films in the system. |
| Digital type | Contains information of all digital type in the system. |
| Group Cinema | Contains information of all group cinema in the system. |
| Partner Account | Contains information of all partner account |
| Cinema | Contains a list of cinemas in the system |
| Cinema Manager | Contains information of all cinema manager account |
| Show Time | Contains a list of show time in the system. |
| Room | Contains information of all room |
| Seat | Contains information of all seat |
| Type of seat | Contains all information of type of seat |
| Ticket | Contains all information of ticket |
| Booking Ticket | Contains a list of order made by users. |
| ShowTime | Contains all time of movie. |
| Customer | Contains information of all customer |
| User Account | Contains information of all user account. |
| Table 59: Entities Description | |

* + - 1. User

**Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| userId | Unique id of user. |
| userPassword | Password of each user. |
| phone | User’s phone number. |
| email | Email of user. |
| Table 60: User class attributes explanation | |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| Getter | Get attribute value |
| Setter | Set value of attribute |
| Table 61: User class methods explanation | |

* + - 1. Admin

**Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| adminId | Unique id of admin. |
| adminPassword | Password of admin. |
| phone | Admin’s phone number. |
| email | Email of admin. |
| Table 62: Admin class attributes explanation | |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| Getter | Get attribute value |
| Setter | Set value of attribute |
| Table 63: Admin method methods explanation | |

* + - 1. Customer

**Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| customerId | Unique id of customer. |
| userId | Id of the account the customer has. |
| phone | Phone number of customer. |
| email | Email of customer. |
| Table 64: Customer class attributes explanation | |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| Getter | Get attribute value |
| Setter | Set value of attribute |
| Table 65: Customer method methods explanation | |

* + - 1. Partner Account

**Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| partnerId | Unique id of partner. |
| partnerPassword | Password of partner. |
| partnerName | Name of partner |
| phone | Phone number of each partner. |
| email | Email of each partner. |
| IsAvailable | Describe that partner is still work or not. |
| Table 66: Partner class attributes explanation | |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| Getter | Get attribute value |
| Setter | Set value of attribute |
| Table 67: Partner class methods explanation | |

* + - 1. Cinema Manager

**Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| managerId | Unique id of cinema manager. |
| managerPassword | Password of cinema manager. |
| managerName | Name of cinema manager |
| phone | Phone number of cinema manager. |
| email | Email of cinema manager. |
| isAvailable | Status of cinema manager |
| Table 68: Cinema Manager class attributes explanation | |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| Getter | Get attribute value |
| Setter | Set value of attribute |
| Table 69: Cinema Manager class methods explanation | |

* + - 1. Film

**Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| filmId | Unique id of film. |
| name | Film’s name. |
| dateRelease | Film’s release date. |
| restricted | Film’s age restriction. |
| filmLength | Film’s length. |
| imdb | Film’s imdb score. |
| digTypeId | Film’s shown in 2D or 3D |
| author | Film’s author |
| movieGenre | Film’s genre |
| filmContent | Film’s summary |
| actorList | Film’s list of actors |
| countries | Countries that the film is shown. |
| trailerLink | Film’s trailer |
| posterPicture | Film’s poster |
| additionPicture | Film’s additional picture |
| filmStatus | Film’s status of whether it is not shown anymore, being shown or is coming in the future. |
| ticketSold | Number of the film’s tickets sold |
| Table 70: Film class attributes explanation | |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| Getter | Get attribute value |
| Setter | Set value of attribute |
| Table 71: Film class methods explanation | |

* + - 1. Movie schedule

**Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| scheduleId | Unique id of movie schedule. |
| scheduleDate | Date of the movie schedule. |
| Table 72: Movie class attributes explanation | |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| Getter | Get attribute value |
| Setter | Set value of attribute |
| Table 73: Movie class methods explanation | |

* + - 1. Show Time

**Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| timeId | Unique id of the time the film is shown. |
| startTime | Time when the movie start. |
| endTime | Time when the movie end. |
| startTimeDouble | Time when the movie start. |
| Table 73: Show time class attributes | |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| Getter | Get attribute value |
| Setter | Set value of attribute |
| Table 74: Show time class method explanation | |

* + - 1. Room

**Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| roomId | Unique room’s id. |
| cinemaId | Id of the cinema the room belong to. |
| capacity | Maximum number of customer the room can take in. |
| name | Room’s name. |
| matrixSizeX | Room’s width. |
| matrixSizeY | Room’s Length. |
| Table 75: Room class attributes explanation | |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| Getter | Get attribute value |
| Setter | Set value of attribute |
| Table 76: Room class methods explanation | |

* + - 1. Digital Type

**Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| digTypeId | Unique id of digital type. |
| name | Type of digital (2D,3D,iMax). |
| Table 77: Digital type class attributes explanation | |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| Getter | Get attribute value |
| Setter | Set value of attribute |
| Table 78: Digital Type class methods explanation | |

* + - 1. Seat

**Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| seatId | Unique Seat’s id. |
| px | Seat’s horizontal pixel. |
| py | Seat’s vertical pixel. |
| locationX | Seat’s horizontal location. |
| locationY | Seat’s vertical location. |
| Table 79: Seat class attributes explanation | |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| Getter | Get attribute value |
| Setter | Set value of attribute |
| Table 80: Seat class methods explanation | |

* + - 1. Type of Seat

**Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| typeSeatId | Unique type of seat’s id. |
| typeName | Name of the type of seat. |
| capacity | Maximum number of customer the type of seat can take in. |
| groupId | Id of the group of cinema the type of seat belongs to. |
| isPrimary | Explant that is this price is the base price. |
| price | Type of seat’s price. |
| Table 81: Type of Seat class attributes explanation | |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| Getter | Get attribute value |
| Setter | Set value of attribute |
| Table 82: Type of Seat class methods explanation | |

* + - 1. Group Cinema

**Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| groupId | Unique Id of the group of cinema. |
| logoImg | Logo of the group of cinema. |
| name | Name of the group of cinema. |
| address | Address of the group of cinema. |
| email | Mail of the group of cinema. |
| phone | Phone of the group of cinema. |
| Table 83: Group Cinema class attributes explanation | |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| Getter | Get attribute value |
| Setter | Set value of attribute |
| Table 84: Group Cinema class methods explanation | |

* + - 1. Cinema

**Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| cinemaId | Unique Cinema id. |
| cinemaName | Cinema’s name. |
| profilePicture | Cinema’s profile picture. |
| cinemaAddress | Cinema’s address. |
| phone | Cinema’s phone number. |
| email | Cinema’s email. |
| openTime | The time when cinema open. |
| introduction | Cinema’s introduction. |
| Table 85: Cinema class attributes explanation | |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| Getter | Get attribute value |
| Setter | Set value of attribute |
| Table 86: Cinema class methods explanation | |

* + - 1. Ticket

**Attribute**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| ticketId | Unique Ticket id. |
| paymentCode | Code of the payment. |
| qrCode | QR’s code. |
| ticketStatus | Ticket’s status. |
| resellDecription | Decription info ticket resell |
| price | Ticket’s price. |
| Table 89: Ticket class attributes explanation | |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| Getter | Get attribute value |
| Setter | Set value of attribute |
| Table 90: Ticket class methods explanation | |

* + - 1. Booking Ticket

**Attribute**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| bookingId | Unique id of the transaction when the customer is booking a ticket. |
| paymentMethodId | Id of the method the customer is paying. |
| paymentCode | Code of the payment. |
| qrCode | QR’s code. |
| quantity | Quantity of how many tickets the customer is buying. |
| bookingDate | The date the customer book the ticket(s). |
| Table 91: Booking ticket class attributes explanation | |

**Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| Getter | Get attribute value |
| Setter | Set value of attribute |
| Table 92: Booking ticket class methods explanation | |

* 1. Algorithms
     1. Display physical cinema room in 2D interface.
        1. Define problem.

It is difficult to transform the physical cinema model into a web interface with easy viewing and control. The interface will help manager and cinema customer easy to interact with the system. Manager can easily add and arrange seat in cinema and cinema customer will easily choose seat for booking ticket.

* + - 1. Solution.

To start create matrix for cinema room, we ask user for input the room capacity. Room's capacity contains from 80 to 300 seats. Our final goal is creating a 2D interface to display a room with all of seats.

First of all, we calculate size of matrix base on given capacity. The system will automatically generate the template for adding seats.

Next, we click on the matrix sell on template to add seat. Each sell will be considered as one seat. User can choose mode to add. If mode is multiple select, user just need to specify the start cell and end cell, all cells in space between the two cells will be selected. If select mode is single select, the node you clicked on will be the node that you select, click it again to deselect.

If user wants more space, user can click extra space part. User click plus row, one more row will add for the matrix template. Similar that, user click plus column, one column will be added. Note that, the max row and column user can increase base X-dimension +3 rows and base Y-dimension +3 columns.

Then, we click preview to see how the room will be numbered and then click save. The room information will be saved to database. Room information will contain capacity, X-dimension, Y-dimension and the array contain all of seat location data.

* + 1. Select group of seat on room’s matrix.
       1. Define problem.

It is difficult for the user to add each chair to a movie theater, especially when the movie theater has more than 150 seats. We offer a zoning solution to select all seats in the selected area.

* + - 1. Solution.

User must take two nodes before fill seat to template for this area inside of top-left point and right-bottom point. First of all, system will count the times of click on the template, start by zero. If count click is odd, mean user just click once, we save this point to start point. If count click is even means user click twice, we save clicked point to end point. Base on start point and end point we calculate the top-left point and bottom-right point. If user selects top-right or some bottom-left point first we always can convert it in to top-left point and bottom-right point. We do that for select or deselect all cell in the rectangle has the top-left point and bottom-right point that we calculated.

We will explain all case that we convert start point and end point to become top-left point and bottom-right point as below.

In case location x of start point greater and location x of end point and location y of start point less than location y of end point. It means that start point now is top-right and end point is bottom-left. We exchange location x of start point and end point for each other.

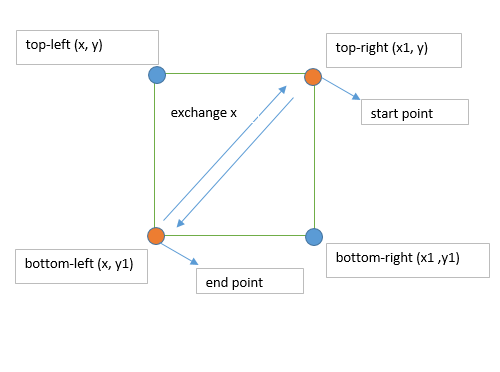


Figure 79: Algorithms - Select group of seat on room's matrix - *a*

In case location y of start point is greater than location y of end point and location x of start point is less than location x of end point. It means start point now is bottom-left point and end point is top-right point. We exchange location x of start point and end point for each other.

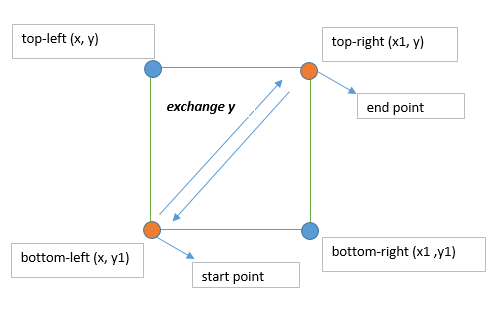


Figure 80: Algorithms - Select group of seat on room's matrix - *b*

In case location x of start point is greater than location x of end point and location y of start point is greater than location y of end point. It means start point now is bottom-right point and end point is top-left point. We exchange swap two point for each other.

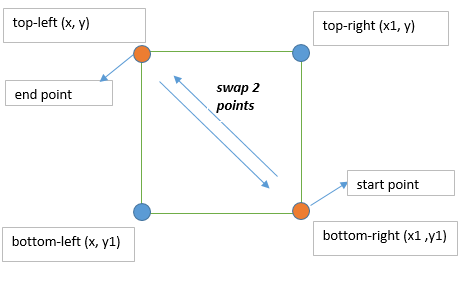


Figure 81: Algorithms - Select group of seat on room's matrix - *c*

In case location x of start point is equal to location x of end point and location y of start point is greater than location y of end point. In this situation user just select one vertical line so start point now is bottom point and end point now is top point. We swap 2 points to each other.

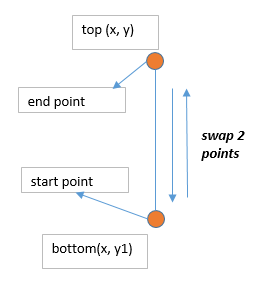


Figure 82: Algorithms - Select group of seat on room's matrix - *d*

In case location x of start point greater than location x of end point and location y of start point equal to location y of end point. In this situation user just select one horizontal line so start point now is right point and end point is left point. We swap 2 points to each other.

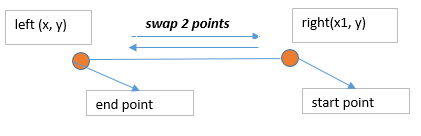


Figure 83: Algorithms - Select group of seat on room's matrix - *e*

* + 1. Movie recommendations for users.
       1. Define problem.

The purpose is to encourage users to buy tickets and increase revenue for partners. We created a movie rating system. From that rating system we suggest the suitable movie for the user.

* + - 1. Solution.

We set up a movie rating system based on four main criteria. The first is the hot of the movie, the new movie or have been shown a time already. Second is the average number of tickets sold each day. Third is based on quality, we rely on imdb points to evaluate this criterion. The final movie rating will be calculated as the sum of all the above criteria multiplied by its weight.

For the first criterion and the third criterion we set the weight of 1 and the first set the weight of 2. Because our ultimate goal is still to increase ticket sales so ticket rank will have higher priority.

For the first criterion, we first calculate the time from the beginning of the film to the present time. If duration is less than 3 we add 3 points if duration is greater than or equal to 3 and less than 6 plus 2 points, if duration is greater than 6 plus 1 point. Taking the calculated points multiplied by the weights scored for the first criterion.

For the second criterion, we first calculate the average number of tickets sold per day. If the average number of tickets sold is less than 100 we add 0 points, if greater than 100 and less than 200 plus one point, if greater than 200 and less than 300 plus two points, if greater than 300 plus 3 points. Taking the calculated points multiplied by the weights scored for the second criterion.

For the third criterion, we rely on the imdb point for quality evaluation. Because sometimes there are some movies without imdb points because of lack of information or Vietnamese film so there is no imdb point. So we do not focus on this criterion very much and just weight it to the lowest level. If imdb score is greater than 7 we add 2 points.

1. System Implementation & Test
   1. Introduction
      1. Overview

This section describes the approach and methodologies used by our group to plan, organize and manage the testing of OCBS system. It provides in detail all necessary information about the implementation and testing procedure of the system included test plans, test cases, test result, test environments, pass/fail criteria and risks estimations as well as a checklist to cover all possible cases.

* + 1. Test Approach
* Goal: Test all features in the whole OCBS system based on the core flow.
* Method: black-box testing
* Technique: check list
  1. Database Relationship Diagram
     1. Physical Diagram



Figure 84: Physical Diagram

* + 1. Data Dictionary

|  |  |
| --- | --- |
| **Entity Data Dictionary: describe contents of all entities** | |
| **Entity Name** | **Description** |
| Film | Contains information of all films in the system. |
| Digital type | Contains information of all digital type in the system. |
| Group Cinema | Contains information of all group cinema in the system. |
| Partner account | Contains information of all partner account |
| Cinema | Contains a list of cinemas in the system |
| Cinema Manager | Contains information of all cinema manager account |
| Show Time | Contains a list of show time in the system. |
| Room | Contains information of all room |
| Seat | Contains information of all seat |
| Type of seat | Contains all information of type of seat |
| Ticket | Contains all information of ticket |
| Booking Ticket | Contains a list of order made by users. |
| News | Contains a list news of cinema. |
| ShowTime | Contains all time of movie. |
| Customer | Contains information of all customer |
| User Account | Contains information of all user account. |
| Table 93: Entity Data Dictionary | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Table Name | Attributes | Description | Domain | Null |
| User Account | userId(PK) | Unique id of user. | String | No |
| userPassword | Password of each user. | String | Yes |
| phone | User’s phone number. | String | Yes |
| email | Email of user. | String | Yes |
| tix |  | float | Yes |
| Admin Account | Id(PK) | Unique id of admin. | Integer | No |
| password | Password of admin. | String | Yes |
| phone | Phone number of admin. | String | Yes |
| email | Email of admin. | String | Yes |
| Customer | customerId(PK) | Unique id of customer. | Integer | No |
| userId | Id of the account the customer has. | String | Yes |
| phone | Phone number of customer. | String | Yes |
| email | Email of customer. | String | Yes |
| Cinema Manager | managerId(PK) | Unique id of cinema manager. | String | No |
| managerPassword | Password of cinema manager. | String | Yes |
| phone | Phone number of cinema manager. | String | Yes |
| email | Email of cinema manager. | String | Yes |
| cinemaId | Id of the cinema manager belong to. | Integer | Yes |
| IsAvailable |  | bit | Yes |
| Film | filmId(PK) | Unique id of film. | Integer | No |
| name | Film’s name. | String | Yes |
| dateRelease | Film’s release date. | Datetime | No |
| restricted | Film’s age restriction. | Integer | Yes |
| filmLength | Film’s length. | Integer | Yes |
| imdb | Film’s imdb score. | Float | Yes |
| digTypeId | Film’s shown in 2D or 3D | String | Yes |
| author | Film’s author | String | Yes |
| movieGenre | Film’s genre | String | Yes |
| filmContent | Film’s summary | String | Yes |
| actorList | Film’s list of actors | String | Yes |
| countries | Countries that the film is shown. | String | Yes |
| trailerLink | Film’s trailer | String | Yes |
| posterPicture | Film’s poster | String | Yes |
| additionPicture | Film’s additional picture | String | Yes |
| filmStatus | Film’s status of whether it is not shown anymore, being shown or is coming in the future. | Integer | Yes |
| Movie Schedule | scheduleId(PK) | Unique id of movie schedule. | Integer | No |
| filmId | Id of film that is shown on the movie schedule. | Integer | Yes |
| timeId | Id of the time the film is shown | Integer | Yes |
| roomId | Id of the room the film is shown in. | Integer | Yes |
| scheduleDate | Date of the movie schedule. | Date | Yes |
| Show time | timeId(PK) | Unique id of the time the film is shown. | Integer | No |
| startTime | Time when the movie start. | String | Yes |
| endTime | Time when the movie end. | String | Yes |
| Room | roomId(PK) | Unique room’s id. | Integer | No |
| cinemaId | Id of the cinema the room belong to. | Integer | Yes |
| capacity | Maximum number of customer the room can take in. | Integer | Yes |
| name | Room’s name. | String | Yes |
| digTypeId | Room is a 2D room, 3D room or iMax room | Integer | Yes |
| matrixSizeX | Room’s width. | Integer | Yes |
| matrixSizeY | Room’s Length. | Integer | Yes |
| Digital Type | digTypeId(PK) | Unique id of digital type. | Integer | No |
| name | Type of digital(2D,3D,iMax). | String | Yes |
| Seat | seatId(PK) | Unique Seat’s id. | Integer | No |
| typeSeatId | Id of the type of seat. | Integer | Yes |
| roomId | Id of the room that have the seat. | Integer | Yes |
| px | Seat’s horizontal pixel. | Integer | Yes |
| py | Seat’s vertical pixel. | Integer | Yes |
| locationX | Seat’s horizontal location. | Integer | Yes |
| locationY | Seat’s vertical location. | Integer | Yes |
| Type of Seat | typeSeatId(PK) | Unique type of seat’s id. | Integer | No |
| typeSeatName | Name of the type of seat. | Integer | Yes |
| capacity | Maximum number of customer the type of seat can take in. | Integer | Yes |
| groupId | Id of the group of cinema the type of seat belong to. | Integer | Yes |
| price | Type of seat’s price. | Float | Yes |
| Group Cinema | groupId(PK) | Unique Id of the group of cinema. | Integer | No |
| logoImg | Logo of the group of cinema. | String | Yes |
| name | Name of the group of cinema. | String | Yes |
| mail | Mail of the group of cinema. | String | Yes |
| address | Address of the group of cinema. | String | Yes |
| phone | Phone of the group of cinema. | String | Yes |
| Cinema | cinemaId(PK) | Unique Cinema id. | Integer | No |
| cinemaName | Cinema’s name. | String | Yes |
| groupId | Id of the group of cinema the cinema belong to. | Integer | Yes |
| profilePicture | Cinema’s profile picture. | String | Yes |
| cinemaAddress | Cinema’s address. | String | Yes |
| phone | Cinema’s phone number. | String | Yes |
| email | Cinema’s email. | String | Yes |
| openTime | The time when cinema open. | String | Yes |
| introduction | Cinema’s introduction. | String | Yes |
| Ticket | ticketId(PK) | Unique Ticket id. | Integer | No |
| bookingId | Id of the booking that is on the ticket. | Integer | Yes |
| scheduleId | id of the transaction when the customer is booking a ticket. | Integer | Yes |
| seatId | Id of the seat that is on the ticket. | Integer | Yes |
| paymentCode | Code of the payment. | String | Yes |
| qrCode | QR’s code. | String | Yes |
| ticketStatus | Ticket’s status. | String | Yes |
| resellDecription | Decription info ticket resell | String | Yes |
| price | Ticket’s price. | float | Yes |
| Booking Ticket | bookingId(Pk) | Unique id of the transaction when the customer is booking a ticket. | Integer | No |
| customerId | Id of the customer. | Integer | Yes |
| paymentMethodId | Id of the method the customer is paying. | Integer | Yes |
| paymentCode | Code of the payment. | String | Yes |
| qrCode | QR’s code. | String | Yes |
| quantity | Quantity of how many tickets the customer is buying. | Integer | Yes |
| bookingDate | The date the customer book the ticket(s). | Datetime | Yes |
| Table 94: Entity Data Dictionary | | | | |

* 1. Implementation
     1. Backend Architecture Implementation

Backend Architecture

*Figure 85: Backend Architecture Implementation*

|  |  |
| --- | --- |
| **Name** | **Description** |
| Clarified | Image Recognition Service |
| Query | Contains complex queries |
| Active Record | Object relational mapping |
| Model | Business Logic Layer |
| Controller | Receives input from client |
| View | Outputs the display of information |
| Client | Web application users |
| Database | Contains a collection of schemas, table, queries |
| Table 95: *Backend Architecture Implementation* | |

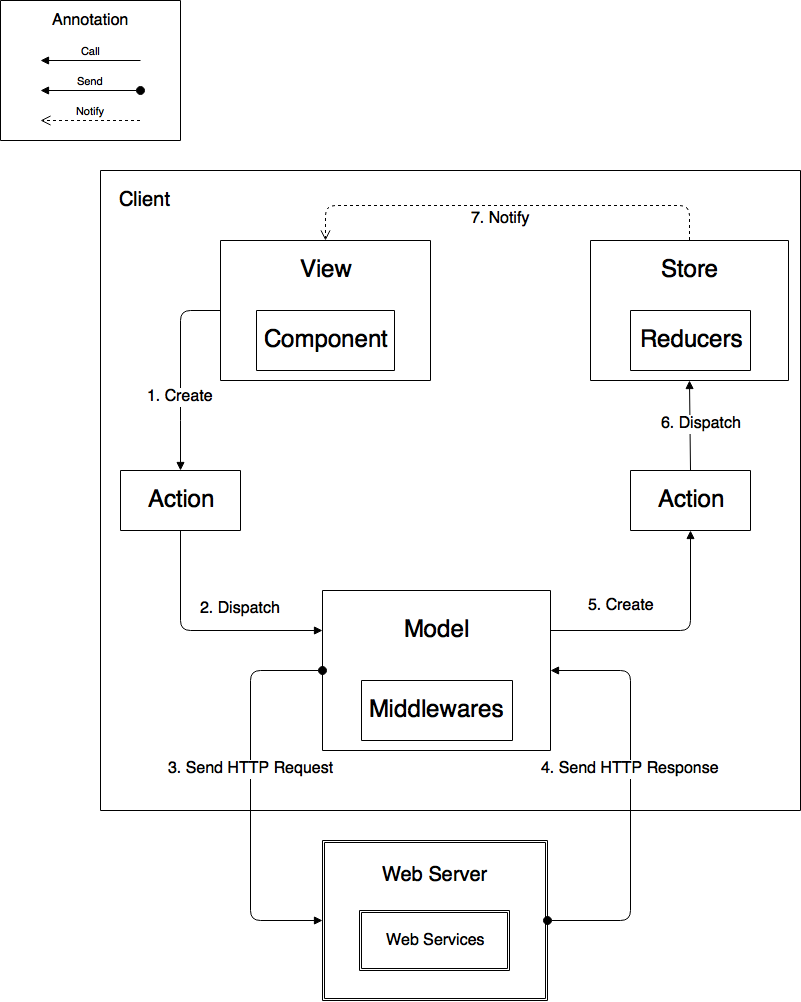
* + 1. Mobile Application Architecture Implementation

Mobile Architecture

*Figure 86: Mobile Application Architecture Implementation*

|  |  |
| --- | --- |
| **Name** | **Description** |
| Controller | Control and handle event of a screen |
| View Model | View properties and commands |
| Model | Business Logic Layer |
| Web Services | Interface to communicate with web application |
| View | Outputs the display of information |
| Table 96: *Mobile Application Architecture Implementation* | |

* + 1. Web Architecture Implementation



*Figure 87: Web Architecture Implementation*

|  |  |
| --- | --- |
| **Name** | **Description** |
| View | Output display of information |
| Components | Encapsulates and reuses part of View |
| Store | Contains application states |
| Reducers | Are functions to manipulate states |
| Action | Are commands to communicate with Model and Store |
| Model | Data Access Layer |
| Middleware | Contains business logic |
| Web Server | Contains Web services |
| Web Services | Interface to communicate with web application |
| Table 97: *Web Architecture Implementation* | |

* + 1. Test Driven Development

We will use Test Driven Development (TDD) to implement web services and backend logic. We choose TDD for couple of reason:

- To protect our implementation against future breaks

- To make sure we understand the requirement before writing any line of code

- To do quality assurance to our code base

- Applying TDD brings confidence when we do refactoring our code base

The test result could be found [here](https://drive.google.com/open?id=0B0EgBSrhB9kfOHhfMUZ1Yk02U28)

* 1. Test Plan
     1. Communication Diagram
        1. User

User purchase ticket

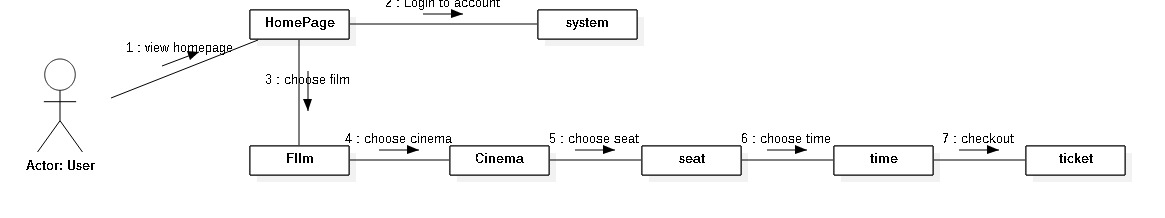


Figure 88: Communication Diagram – User purchase ticket

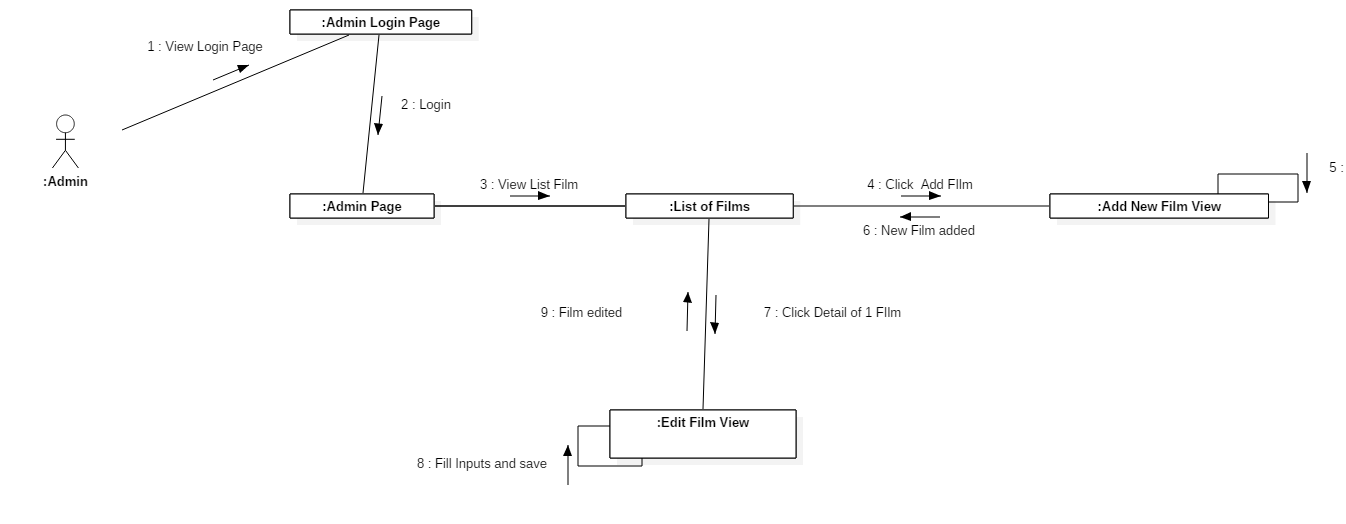
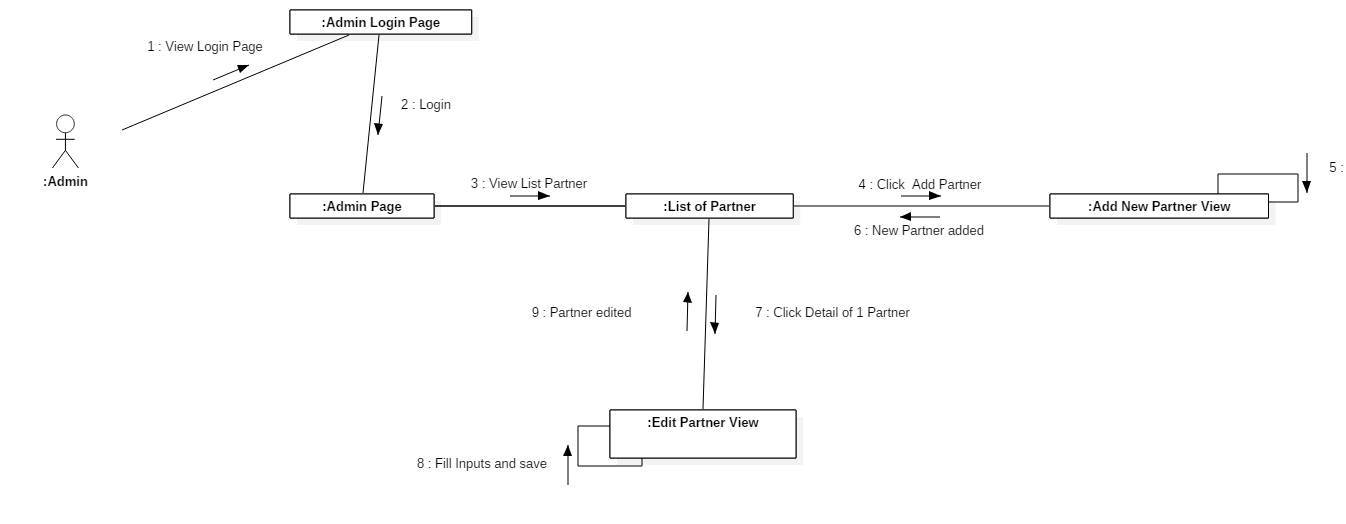
* + - 1. Admin
         1. Admin Manage Film

Figure 89: Communication Diagram – Admin- Admin Manage Film

* + - * 1. Admin Manage Partner

Figure 90: Communication Diagram – Admin- Admin Manage Partner

* + - * 1. Admin Manage Group of Cinema

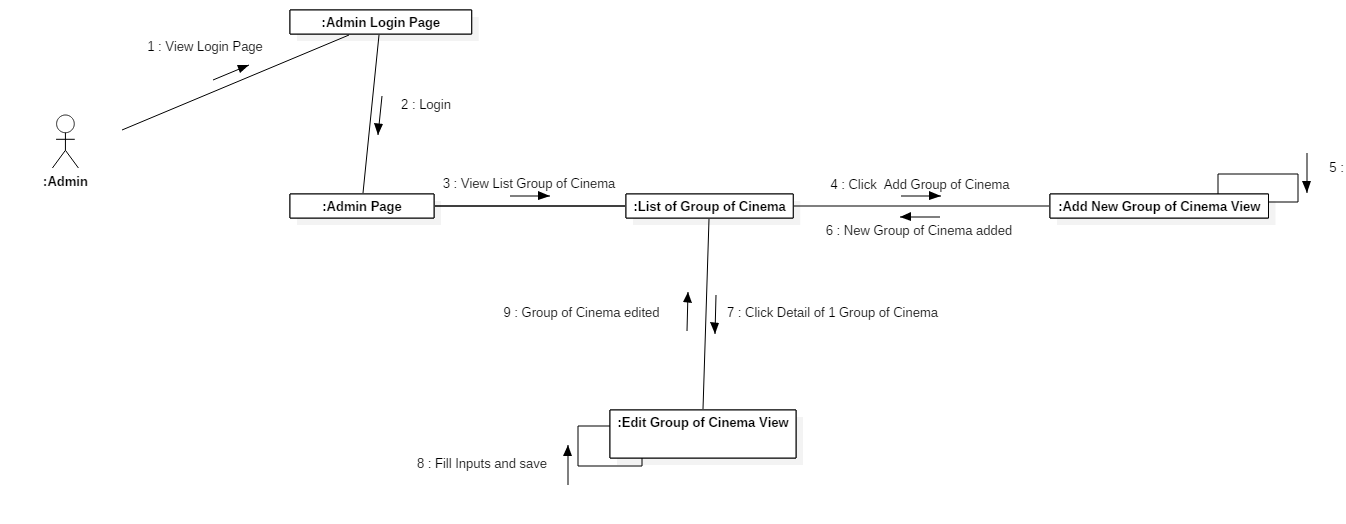


Figure 91: Communication Diagram – Admin- Admin Manage Group of Cinema

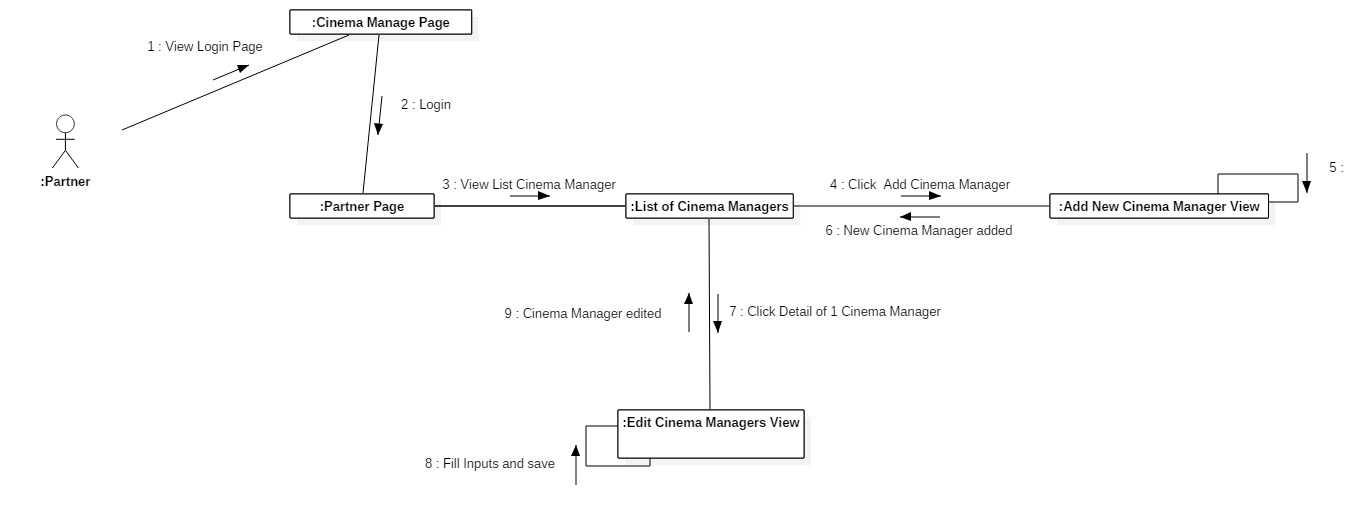
* + - 1. Partner
         1. Partner Manage Cinema

Figure 92: Communication Diagram – Partner - Partner Manage Cinema

* + - * 1. Partner Add Room

Figure 93: Communication Diagram – Partner - Partner Add Room

* + - 1. Cinema Manager

Cinema Manager Manage Schedule

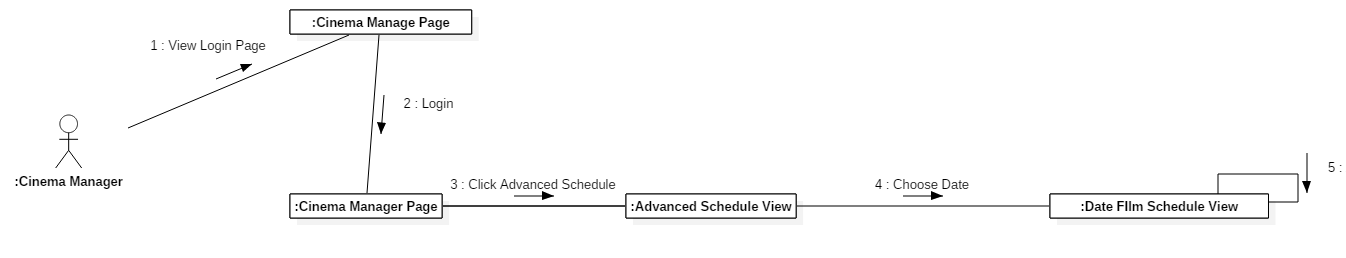


Figure 94: Communication Diagram – Cinema Manager – Cinema Manager Manage Schedule

* 1. System Testing Test Case
     1. Web Application Test Cases
        1. <Customer> View Film Detail

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **SVFD01** | View detail of film by clicking on the picture of the film. | 1. Go to home page.  2. Click on the picture of the film. | Web application go t o the film detail page | Pass: 47  Fail: 3 | From 1/3/2017  To 7/3/2017 |
| **SVFD02** | View detail of film by clicking on the button “Mua Vé” under the picture of the film. | 1. Go to home page.  2. Hover on the picture of the film  3. Click on the button “Mua Vé” under the picture of the film | Web application go t o the film detail page. | Pass: 9  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| *Table 98: Web application test case- <Customer> View Film Detail* | | | | | |

* + - 1. <Customer> Choose Tickets

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **SCT01** | Choose ticket by clicking on picture of the film at home page. | 1. Go to home page.  2. Click on film’s picture.  3. Choose group of cinema and date to watch the film.  4. Choose the film’s showtime.  5. Choose the amount of seats to buy.  6.Click on button “Chọn Ghế”. | Web application goes to choose seat and payment page. | Pass: 40  Fail: 10 | From 20/7/2018  To 28/7/2018 |
| **SCT02** | Choose ticket by clicking on the button “Mua Vé” under the picture of the film at home page. | 1. Go to home page.  2. Hover on the picture of the film  3. Click on the button “Mua Vé” under the picture of the film.  4. Choose group of cinema and date to watch the film.  5. Choose the film’s showtime.  6. Choose the amount of seats to buy.  7Click on button “Chọn Ghế”. | Web application goes to choose seat and payment page. | Pass: 28  Fail: 2 | From 1/3/2017  To 7/3/2017 |
| **SCT03** | Choose ticket by clicking on picture of the film at home page and choose 0 seat. | 1. Go to home page.  2. Click on film’s picture.  3. Choose group of cinema and date to watch the film.  4. Choose the film’s showtime.  5. Set the amount of seats to buy to 0.  6.Click on button “Chọn Ghế”. | A popup message will appear which says “Bạn chưa chọn ghế nào!”. | Pass: 40  Fail: 10 | From 20/7/2018  To 28/7/2018 |
| **SCT04** | Choose ticket by clicking on the button “Mua Vé” under the picture of the film at home page and choose 0 seat. | 1. Go to home page.  2. Hover on the picture of the film  3. Click on the button “Mua Vé” under the picture of the film.  4. Choose group of cinema and date to watch the film.  5. Choose the film’s showtime.  6. Set the amount of seats to buy to 0.  7.Click on button “Chọn Ghế”. | A popup message will appear which says “Bạn chưa chọn ghế nào!”. | Pass: 40  Fail: 10 | From 20/7/2018  To 28/7/2018 |
| **SCT05** | Choose ticket by clicking on picture of the film at home page and choose more than 10 seats. | 1. Go to home page.  2. Click on film’s picture.  3. Choose group of cinema and date to watch the film.  4. Choose the film’s showtime.  5. Set the amount of seats to buy to more than 10.  6.Click on button “Chọn Ghế”. | A popup message will appear which says “Bạn Chỉ Được Mua 10 Vé!”. | Pass: 40  Fail: 10 | From 20/7/2018  To 28/7/2018 |
| **SCT04** | Choose ticket by clicking on the button “Mua Vé” under the picture of the film at home page and choose more than 10 seats. | 1. Go to home page.  2. Hover on the picture of the film  3. Click on the button “Mua Vé” under the picture of the film.  4. Choose group of cinema and date to watch the film.  5. Choose the film’s showtime.  6. Set the amount of seats to buy to more than 10.  7.Click on button “Chọn Ghế”. | A popup message will appear which says “Bạn Chỉ Được Mua 10 Vé!”. | Pass: 40  Fail: 10 | From 20/7/2018  To 28/7/2018 |
| *Table 99: Web application test case- <Customer> Choose Tickets* | | | | | |

* + - 1. <Customer> Watch Film’s Trailer

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **SWFT01** | Watch film’s trailer on home page. | 1. Go t home page.  2. Hover on the picture of the film.  3. Click on the play button in the center of the picture | A video popup will appear which contains the trailer of the corresponding film. | Pass: 45  Fail: 5 | From 1/3/2017  To 7/3/2017 |
| **SWFT02** | Watch film’s trailer on home page. | 1. Go t home page.  2. Hover on the picture of the film.  3. Click on the button “Mua Vé” under the picture of the film.  4. Hover on the picture of the film.  5. Click on the play button in the center of the picture | A video popup will appear which contains the trailer of the corresponding film. | Pass: 8  Fail: 2 | From 1/3/2017  To 7/3/2017 |
| *Table 100: Web application test case- <Customer> Watch Film’s Trailer* | | | | | |

* + - 1. <Customer> Choose Seat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **SCS01** | Choose ticket by clicking on picture of the film at home page.. | 1. Go to home page.  2. Click on film’s picture.  3. Choose group of cinema and date to watch the film.  4. Choose the film’s showtime.  5. Choose the amount of tickets to buy.  6.Click on button “Chọn Ghế”.  7. Click on the desired seats button.  8. Fill in Email and Phone number.  9.Click on the “Đặt Vé” button | A ticket confim popup will appear which contains the seats chosen. | Pass: 25  Fail: 5 | From 1/3/2017  To 7/3/2017 |
| **SCS02** | Choose seat by clicking on the button “Mua Vé” under the picture of the film. | 1. Go to home page.  2. Hover on the picture of the film  3. Click on the button “Mua Vé” under the picture of the film.  4. Choose group of cinema and date to watch the film.  5. Choose the film’s showtime.  6. Choose the amount of tickets to buy.  7.Click on button “Chọn Ghế”.  8. Click on the desired seats button.  9. Fill in Email and Phone number.  10.Click on the “Đặt Vé” button | A ticket confim popup will appear which contains the seats chosen. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **SCS03** | Choose ticket by clicking on picture of the film at home page and fill in the email input incorrectly. | 1. Go to home page.  2. Click on film’s picture.  3. Choose group of cinema and date to watch the film.  4. Choose the film’s showtime.  5. Choose the amount of tickets to buy.  6.Click on button “Chọn Ghế”.  7. Click on the desired seats button.  8. In the Email input fill in an email that is wrong format.  9. Fill in Phone number.  10.Click on the “Đặt Vé” button | A popup message will appear which says “Email và phone sai format!”. | Pass: 25  Fail: 5 | From 1/3/2017  To 7/3/2017 |
| **SCS04** | Choose seat by clicking on the button “Mua Vé” under the picture of the film at home page and fill in the email input incorrectly.. | 1. Go to home page.  2. Hover on the picture of the film  3. Click on the button “Mua Vé” under the picture of the film.  4. Choose group of cinema and date to watch the film.  5. Choose the film’s showtime.  6. Choose the amount of tickets to buy.  7.Click on button “Chọn Ghế”.  8. Click on the desired seats button.  9. In the Email input fill in an email that is wrong format.  10. Fill in Phone number  11.Click on the “Đặt Vé” button | A ticket confim popup will appear which contains the seats chosen. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **SCS05** | Choose ticket by clicking on picture of the film at home page and fill in the phone input incorrectly. | 1. Go to home page.  2. Click on film’s picture.  3. Choose group of cinema and date to watch the film.  4. Choose the film’s showtime.  5. Choose the amount of tickets to buy.  6.Click on button “Chọn Ghế”.  7. Click on the desired seats button.  8. In the Phone input fill in an phone number that is less than 10 digits or more than 11 digits.  9. Fill in Email address.  10.Click on the “Đặt Vé” button | A popup message will appear which says “Email và phone sai format!”. | Pass: 25  Fail: 5 | From 1/3/2017  To 7/3/2017 |
| **SCS06** | Choose seat by clicking on the button “Mua Vé” under the picture of the film at home page and fill in the phone input incorrectly.. | 1. Go to home page.  2. Hover on the picture of the film  3. Click on the button “Mua Vé” under the picture of the film.  4. Choose group of cinema and date to watch the film.  5. Choose the film’s showtime.  6. Choose the amount of tickets to buy.  7.Click on button “Chọn Ghế”.  8. Click on the desired seats button.  9. In the Phone input fill in an phone number that is less than 10 digits or more than 11 digits.  10. Fill in Email address  11.Click on the “Đặt Vé” button | A popup message will appear which says “Email và phone sai format!”. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **SCS07** | Choose ticket by clicking on picture of the film at home page and choose seats with empty seat between chosen seats. | 1. Go to home page.  2. Click on film’s picture.  3. Choose group of cinema and date to watch the film.  4. Choose the film’s showtime.  5. Choose more than 2 tickets to buy.  6.Click on button “Chọn Ghế”.  7. Click on 1 seat button and then click on another seat with an empty seat between the 1st and 2nd chosen seat. | A popup message will appear which says “Không để ghế trống ở giữa”. | Pass: 25  Fail: 5 | From 1/3/2017  To 7/3/2017 |
| **SCS08** | Choose ticket by clicking on picture of the film at home page and choose seats with empty seat between chosen seats. | 1. Go to home page.  2. Click on film’s picture.  3. Choose group of cinema and date to watch the film.  4. Choose the film’s showtime.  5. Choose more than 2 tickets to buy.  6.Click on button “Chọn Ghế”.  7. Click on 1 seat button and then click on another seat with an empty seat between the 1st and 2nd chosen seat.  8. Fill in Email and Phone number  11.Click on the “Đặt Vé” button | A popup message will appear which says “Không để ghế trống ở giữa”. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| *Table 101: Web application test case- <Customer> Choose Seat* | | | | | |

* + - 1. <Customer> Purchase Tickets

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **SPT01** | Purchase ticket by clicking on picture of the film at home page.. | 1. Go to home page.  2. Click on film’s picture.  3. Choose group of cinema and date to watch the film.  4. Choose the film’s showtime.  5. Choose the amount of tickets to buy.  6.Click on button “Chọn Ghế”.  7. Click on the desired seats button.  8. Fill in Email and Phone number.  9.Click on the “Đặt Vé” button  10.Click on the “PayPal Checkout” button on the popup box.  11. Pay using paypal account or credit card.  12. | A ticket confim popup will appear which contains the seats chosen. | Pass: 25  Fail: 5 | From 1/3/2017  To 7/3/2017 |
| **SPT02** | Purchase seat by clicking on the button “Mua Vé” under the picture of the film. | 1. Go to home page.  2. Hover on the picture of the film  3. Click on the button “Mua Vé” under the picture of the film.  4. Choose group of cinema and date to watch the film.  5. Choose the film’s showtime.  6. Choose the amount of tickets to buy.  7.Click on button “Chọn Ghế”.  8. Click on the desired seats button.  9. Fill in Email and Phone number.  10.Click on the “Đặt Vé” button | A ticket confim popup will appear which contains the seats chosen. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **SPT03** | Choose ticket by clicking on picture of the film at home page and fill in the email input incorrectly. | 1. Go to home page.  2. Click on film’s picture.  3. Choose group of cinema and date to watch the film.  4. Choose the film’s showtime.  5. Choose the amount of tickets to buy.  6.Click on button “Chọn Ghế”.  7. Click on the desired seats button.  8. In the Email input fill in an email that is wrong format.  9. Fill in Phone number.  10.Click on the “Đặt Vé” button | A popup message will appear which says “Email và phone sai format!”. | Pass: 25  Fail: 5 | From 1/3/2017  To 7/3/2017 |
| **SPT04** | Choose seat by clicking on the button “Mua Vé” under the picture of the film at home page and fill in the email input incorrectly.. | 1. Go to home page.  2. Hover on the picture of the film  3. Click on the button “Mua Vé” under the picture of the film.  4. Choose group of cinema and date to watch the film.  5. Choose the film’s showtime.  6. Choose the amount of tickets to buy.  7.Click on button “Chọn Ghế”.  8. Click on the desired seats button.  9. In the Email input fill in an email that is wrong format.  10. Fill in Phone number  11.Click on the “Đặt Vé” button | A ticket confim popup will appear which contains the seats chosen. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **SPT05** | Choose ticket by clicking on picture of the film at home page and fill in the phone input incorrectly. | 1. Go to home page.  2. Click on film’s picture.  3. Choose group of cinema and date to watch the film.  4. Choose the film’s showtime.  5. Choose the amount of tickets to buy.  6.Click on button “Chọn Ghế”.  7. Click on the desired seats button.  8. In the Phone input fill in an phone number that is less than 10 digits or more than 11 digits.  9. Fill in Email address.  10.Click on the “Đặt Vé” button | A popup message will appear which says “Email và phone sai format!”. | Pass: 25  Fail: 5 | From 1/3/2017  To 7/3/2017 |
| **SPT06** | Choose seat by clicking on the button “Mua Vé” under the picture of the film at home page and fill in the phone input incorrectly.. | 1. Go to home page.  2. Hover on the picture of the film  3. Click on the button “Mua Vé” under the picture of the film.  4. Choose group of cinema and date to watch the film.  5. Choose the film’s showtime.  6. Choose the amount of tickets to buy.  7.Click on button “Chọn Ghế”.  8. Click on the desired seats button.  9. In the Phone input fill in an phone number that is less than 10 digits or more than 11 digits.  10. Fill in Email address  11.Click on the “Đặt Vé” button | A popup message will appear which says “Email và phone sai format!”. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **SPT07** | Choose ticket by clicking on picture of the film at home page and choose seats with empty seat between chosen seats. | 1. Go to home page.  2. Click on film’s picture.  3. Choose group of cinema and date to watch the film.  4. Choose the film’s showtime.  5. Choose more than 2 tickets to buy.  6.Click on button “Chọn Ghế”.  7. Click on 1 seat button and then click on another seat with an empty seat between the 1st and 2nd chosen seat. | A popup message will appear which says “Không để ghế trống ở giữa”. | Pass: 25  Fail: 5 | From 1/3/2017  To 7/3/2017 |
| **SPT08** | Choose ticket by clicking on picture of the film at home page and choose seats with empty seat between chosen seats. | 1. Go to home page.  2. Click on film’s picture.  3. Choose group of cinema and date to watch the film.  4. Choose the film’s showtime.  5. Choose more than 2 tickets to buy.  6.Click on button “Chọn Ghế”.  7. Click on 1 seat button and then click on another seat with an empty seat between the 1st and 2nd chosen seat.  8. Fill in Email and Phone number  11.Click on the “Đặt Vé” button | A popup message will appear which says “Không để ghế trống ở giữa”. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| *Table 102: Web application test case- <Customer> Purchase Ticket* | | | | | |

* + - 1. <Admin> Add Partner

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **AAP01** | Create an account with role Partner | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Partner Account” button.  2. In List Partner Account view, click “Create Partner Account” button.  3. Input fields:   * **Partner Id**: “KhoiNM, QuanVH, LocTD…”. * **Partner Password**: “abc123,asdadssd, …”. * **Group Cinema**: select one of the options cinnema * **Phone**: 01223222227 * **Email**: abc@gmail.com,...   4. Click “Save changes” button. | Web application shows message: “Create account successfully”. | Pass: 7  Fail: 3 | From 1/3/2017  To 7/3/2017 |
| **AAA02** | Create account without partner id | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Partner Account” button.  2. In List Partner Account view, click “Create Partner Account” button.  3. Input fields:   * **Partner Id**: * **Partner Password**: “abc123,asdadssd, …”. * **Group Cinema**: select one of the options cinnema * **Phone**: 01223222227 * **Email**: abc@gmail.com,...   4. Click “Save changes” button. | Web application shows error message “Required” under Partner Id fields. | Pass: 0  Fail: 4 | From 1/3/2017  To 7/3/2017 |
| **AAA03** | Create account with user name has existed | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Partner Account” button.  2. In List Partner Account view, click “Create Partner Account” button.  3. Input fields:   * **Partner Id**: existed partner id * **Partner Password**: “abc123,asdadssd, …”. * **Group Cinema**: select one of the options cinnema * **Phone**: 01223222227 * **Email**: abc@gmail.com,...   4. Click “Save changes” button. | Web application shows error message “partner id existed”. | Pass: 8  Fail: 2 | From 1/3/2017  To 7/3/2017 |
| **AAA04** | Create account without password | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Partner Account” button.  2. In List Partner Account view, click “Create Partner Account” button.  3. Input fields:   * **Partner Id**: “KhoiNM, QuanVH, LocTD…”. * **Partner Password**: * **Group Cinema**: select one of the options cinnema * **Phone**: 01223222227 * **Email**: abc@gmail.com,...   4. Click “Save changes” button. | Web application shows error message “Required” under Full Password fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **AAA05** | Create account without Group cinema | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Partner Account” button.  2. In List Partner Account view, click “Create Partner Account” button.  3. Input fields:   * **Partner Id**: “KhoiNM, QuanVH, LocTD…”. * **Partner Password**: * **Group Cinema**: * **Phone**: 01223222227 * **Email**: abc@gmail.com,...   4. Click “Save changes” button. | Web application shows error message “Required” under Group cinema fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **AAA06** | Create account without Phone | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Partner Account” button.  2. In List Partner Account view, click “Create Partner Account” button.  3. Input fields:   * **Partner Id**: “KhoiNM, QuanVH, LocTD…”. * **Partner Password**: * **Group Cinema**: select one of the options cinnema * **Email**: abc@gmail.com,...   4. Click “Save changes” button. | Web application shows error message “Required” under Phone fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **AAA07** | Create account without Email | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Partner Account” button.  2. In List Partner Account view, click “Create Partner Account” button.  3. Input fields:   * **Partner Id**: “KhoiNM, QuanVH, LocTD…”. * **Partner Password**: * **Group Cinema**: select one of the options cinnema * **Phone**: 01223222227 * **Email**:   4. Click “Save changes” button. | Web application shows error message “Required” under Email fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| *Table 103: Web application test case- <Admin> Add Partner* | | | | | |

* + - 1. <Admin> Edit Partner

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **ACP01** | Edit an account with role Partner | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Partner Account” button.  2. In List Partner Account view, click “Detail” button.  3. Input fields:   * **Partner Id**: “KhoiNM, QuanVH, LocTD…”. * **Partner Password**: “abc123,asdadssd, …”. * **Group Cinema**: select one of the options cinnema * **Phone**: 01223222227 * **Email**: abc@gmail.com,...   4. Click “Save changes” button. | Web application shows message: “Edit account successfully”. | Pass: 7  Fail: 3 | From 21/7/2018  To 5/8/2018 |
| **ACP02** | Edit account without partner id | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Partner Account” button.  2. In List Partner Account view, click “Detail” button.  3. Input fields:   * **Partner Id**: * **Partner Password**: “abc123,asdadssd, …”. * **Group Cinema**: select one of the options cinnema * **Phone**: 01223222227 * **Email**: abc@gmail.com,...   4. Click “Save changes” button. | Web application shows error message “Required” under Partner Id fields. | Pass: 0  Fail: 4 | From 21/7/2018  To 5/8/2018 |
| **ACP03** | Edit account with user name has existed | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Partner Account” button.  2. In List Partner Account view, click “Detail” button.  3. Input fields:   * **Partner Id**: existed partner id * **Partner Password**: “abc123,asdadssd, …”. * **Group Cinema**: select one of the options cinnema * **Phone**: 01223222227 * **Email**: abc@gmail.com,...   4. Click “Save changes” button. | Web application shows error message “partner id existed”. | Pass: 8  Fail: 2 | From 21/7/2018  To 5/8/2018 |
| **ACP04** | Edit account without password | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Partner Account” button.  2. In List Partner Account view, click “Detail” button.  3. Input fields:   * **Partner Id**: “KhoiNM, QuanVH, LocTD…”. * **Partner Password**: * **Group Cinema**: select one of the options cinnema * **Phone**: 01223222227 * **Email**: abc@gmail.com,...   4. Click “Save changes” button. | Web application shows error message “Required” under Full Password fields. | Pass: 4  Fail: 1 | From 21/7/2018  To 5/8/2018 |
| **ACP05** | Edit account without Group cinema | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Partner Account” button.  2. In List Partner Account view, click “Detail” button.  3. Input fields:   * **Partner Id**: “KhoiNM, QuanVH, LocTD…”. * **Partner Password**: * **Group Cinema**: * **Phone**: 01223222227 * **Email**: abc@gmail.com,...   4. Click “Save changes” button. | Web application shows error message “Required” under Group cinema fields. | Pass: 4  Fail: 1 | From 21/7/2018  To 5/8/2018 |
| **ACP06** | Edit account without Phone | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Partner Account” button.  2. In List Partner Account view, click “Detail” button.  3. Input fields:   * **Partner Id**: “KhoiNM, QuanVH, LocTD…”. * **Partner Password**: * **Group Cinema**: select one of the options cinnema * **Email**: abc@gmail.com,...   4. Click “Save changes” button. | Web application shows error message “Required” under Phone fields. | Pass: 4  Fail: 1 | From 21/7/2018  To 5/8/2018 |
| **ACP07** | Edit account without Email | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Partner Account” button.  2. In List Partner Account view, click “Detail” button.  3. Input fields:   * **Partner Id**: “KhoiNM, QuanVH, LocTD…”. * **Partner Password**: * **Group Cinema**: select one of the options cinnema * **Phone**: 01223222227 * **Email**:   4. Click “Save changes” button. | Web application shows error message “Required” under Email fields. | Pass: 4  Fail: 1 | From 21/7/2018  To 5/8/2018 |
| *Table 104: Web application test case- <Admin> Edit Partner* | | | | | |

* + - 1. <Admin> Delete Partner

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **ACP01** | Delete an account with role Partner | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Partner Account” button.  3. In List Partner Account view, click “Delete” button.  4. Click button “Yes” when asked for confimation | Web application shows the account disappear from the list. | Pass: 7  Fail: 3 | From 21/7/2018  To 5/8/2018 |
| *Table 105: Web application test case- <Admin> Delete Partner* | | | | | |

* + - 1. <Admin> Create Film

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **ACP01** | Add a film | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Film” button.  . In List Film view, click “Create Film” button.  3. Input fields:  - Film name: Lộ Mặt  - Date: 20/7/2018  -Restricted: 16  -Film Length: 110  -Author: Vĩnh Thuyên  -Film Genre: Hài hước, tình cảm.  -Actor List:  Hứa Minh Đạt, Mạc Văn Khoa.  -Countries: Việt Nam  -Trailer Link: <https://www.youtube.com/embed/jxIrPADiMQU>  -Poster Picture: Choose image  -Additional Picture: Choose image  -Film status: Now Showing.  -Film Content: fsgffsdfgddg  4. Click “Save changes” button. | Web application shows message: “Create film successfully”. | Pass: 7  Fail: 3 | From 21/7/2018  To 5/8/2018 |
| **ACP02** | Add a film without Film name | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Film” button.  . In List Film view, click “Create Film” button.  3. Input fields:  - Film name:  - Date: 20/7/2018  -Restricted: 16  -Film Length: 110  -Author: Vĩnh Thuyên  -Film Genre: Hài hước, tình cảm.  -Actor List:  Hứa Minh Đạt, Mạc Văn Khoa.  -Countries: Việt Nam  -Trailer Link: <https://www.youtube.com/embed/jxIrPADiMQU>  -Poster Picture: Choose image  -Additional Picture: Choose image  -Film status: Now Showing.  -Film Content: fsgffsdfgddg  4. Click “Save changes” button. | Web application shows error message “Required” under Film Name fields. | Pass: 0  Fail: 4 | From 21/7/2018  To 5/8/2018 |
| **ACP03** | Add a film with Film name has existed | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Film” button.  . In List Film view, click “Create Film” button.  3. Input fields:  - Film name: Existed film name  - Date: 20/7/2018  -Restricted: 16  -Film Length: 110  -Author: Vĩnh Thuyên  -Film Genre: Hài hước, tình cảm.  -Actor List:  Hứa Minh Đạt, Mạc Văn Khoa.  -Countries: Việt Nam  -Trailer Link: <https://www.youtube.com/embed/jxIrPADiMQU>  -Poster Picture: Choose image  -Additional Picture: Choose image  -Film status: Now Showing.  -Film Content: fsgffsdfgddg  4. Click “Save changes” button. | Web application shows error message “film name existed”. | Pass: 8  Fail: 2 | From 21/7/2018  To 5/8/2018 |
| **ACP04** | Add a film without Date | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Film” button.  . In List Film view, click “Create Film” button.  3. Input fields:  - Film name: Lộ Mặt  -Restricted: 16  -Film Length: 110  -Author: Vĩnh Thuyên  -Film Genre: Hài hước, tình cảm.  -Actor List:  Hứa Minh Đạt, Mạc Văn Khoa.  -Countries: Việt Nam  -Trailer Link: <https://www.youtube.com/embed/jxIrPADiMQU>  -Poster Picture: Choose image  -Additional Picture: Choose image  -Film status: Now Showing.  -Film Content: fsgffsdfgddg  4. Click “Save changes” button. | Web application shows error message “Required” under Date fields. | Pass: 4  Fail: 1 | From 21/7/2018  To 5/8/2018 |
| **ACP05** | Add a film without Author | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Film” button.  . In List Film view, click “Create Film” button.  3. Input fields:  - Film name: Lộ Mặt  - Date: 20/7/2018  -Restricted: 16  -Film Length: 110  -Author:  -Film Genre: Hài hước, tình cảm.  -Actor List:  Hứa Minh Đạt, Mạc Văn Khoa.  -Countries: Việt Nam  -Trailer Link: <https://www.youtube.com/embed/jxIrPADiMQU>  -Poster Picture: Choose image  -Additional Picture: Choose image  -Film status: Now Showing.  -Film Content: fsgffsdfgddg  4. Click “Save changes” button. | Web application shows error message “Required” under Author fields. | Pass: 4  Fail: 1 | From 21/7/2018  To 5/8/2018 |
| **ACP06** | Add a film without Film Genre | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Film” button.  . In List Film view, click “Create Film” button.  3. Input fields:  - Film name: Lộ Mặt  - Date: 20/7/2018  -Restricted: 16  -Film Length: 110  -Author: Vĩnh Thuyên  -Film Genre:  -Actor List:  Hứa Minh Đạt, Mạc Văn Khoa.  -Countries: Việt Nam  -Trailer Link: <https://www.youtube.com/embed/jxIrPADiMQU>  -Poster Picture: Choose image  -Additional Picture: Choose image  -Film status: Now Showing.  -Film Content: fsgffsdfgddg  4. Click “Save changes” button. | Web application shows error message “Required” under Film Genre fields. | Pass: 4  Fail: 1 | From 21/7/2018  To 5/8/2018 |
| **ACP07** | Add a film without Actor List | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Film” button.  . In List Film view, click “Create Film” button.  3. Input fields:  - Film name: Lộ Mặt  - Date: 20/7/2018  -Restricted: 16  -Film Length: 110  -Author: Vĩnh Thuyên  -Film Genre: Hài hước, tình cảm.  -Actor List:  -Countries: Việt Nam  -Trailer Link: <https://www.youtube.com/embed/jxIrPADiMQU>  -Poster Picture: Choose image  -Additional Picture: Choose image  -Film status: Now Showing.  -Film Content: fsgffsdfgddg  4. Click “Save changes” button. | Web application shows error message “Required” under Actor List fields. | Pass: 4  Fail: 1 | From 21/7/2018  To 5/8/2018 |
| *Table 106: Web application test case- <Admin> Create Film* | | | | | |

* + - 1. <Admin> Edit Film

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **ACP01** | Edit a Film | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Film” button.  . In List Film view, click “Detail” button.  3. Input fields:  - Film name: Lộ Mặt  - Date: 20/7/2018  -Restricted: 16  -Film Length: 110  -Author: Vĩnh Thuyên  -Film Genre: Hài hước, tình cảm.  -Actor List:  Hứa Minh Đạt, Mạc Văn Khoa.  -Countries: Việt Nam  -Trailer Link: <https://www.youtube.com/embed/jxIrPADiMQU>  -Poster Picture: Choose image  -Additional Picture: Choose image  -Film status: Now Showing.  -Film Content: fsgffsdfgddg  4. Click “Save changes” button. | Web application shows message: “Edit film successfully”. | Pass: 7  Fail: 3 | From 21/7/2018  To 5/8/2018 |
| **ACP02** | Edit a Film without Film name | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Film” button.  . In List Film view, click “Detail” button.  3. Input fields:  - Film name:  - Date: 20/7/2018  -Restricted: 16  -Film Length: 110  -Author: Vĩnh Thuyên  -Film Genre: Hài hước, tình cảm.  -Actor List:  Hứa Minh Đạt, Mạc Văn Khoa.  -Countries: Việt Nam  -Trailer Link: <https://www.youtube.com/embed/jxIrPADiMQU>  -Poster Picture: Choose image  -Additional Picture: Choose image  -Film status: Now Showing.  -Film Content: fsgffsdfgddg  4. Click “Save changes” button. | Web application shows error message “Required” under Film Name fields. | Pass: 0  Fail: 4 | From 21/7/2018  To 5/8/2018 |
| **ACP03** | Edit a Film with Film name has existed | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Film” button.  . In List Film view, click “Detail” button.  3. Input fields:  - Film name: Existed film name  - Date: 20/7/2018  -Restricted: 16  -Film Length: 110  -Author: Vĩnh Thuyên  -Film Genre: Hài hước, tình cảm.  -Actor List:  Hứa Minh Đạt, Mạc Văn Khoa.  -Countries: Việt Nam  -Trailer Link: <https://www.youtube.com/embed/jxIrPADiMQU>  -Poster Picture: Choose image  -Additional Picture: Choose image  -Film status: Now Showing.  -Film Content: fsgffsdfgddg  4. Click “Save changes” button. | Web application shows error message “film name existed”. | Pass: 8  Fail: 2 | From 21/7/2018  To 5/8/2018 |
| **ACP04** | Edit a Film without Date | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Film” button.  . In List Film view, click “Detail” button.  3. Input fields:  - Film name: Lộ Mặt  -Restricted: 16  -Film Length: 110  -Author: Vĩnh Thuyên  -Film Genre: Hài hước, tình cảm.  -Actor List:  Hứa Minh Đạt, Mạc Văn Khoa.  -Countries: Việt Nam  -Trailer Link: <https://www.youtube.com/embed/jxIrPADiMQU>  -Poster Picture: Choose image  -Additional Picture: Choose image  -Film status: Now Showing.  -Film Content: fsgffsdfgddg  4. Click “Save changes” button. | Web application shows error message “Required” under Date fields. | Pass: 4  Fail: 1 | From 21/7/2018  To 5/8/2018 |
| **ACP05** | Edit a Film without Author | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Film” button.  . In List Film view, click “Detail” button.  3. Input fields:  - Film name: Lộ Mặt  - Date: 20/7/2018  -Restricted: 16  -Film Length: 110  -Author:  -Film Genre: Hài hước, tình cảm.  -Actor List:  Hứa Minh Đạt, Mạc Văn Khoa.  -Countries: Việt Nam  -Trailer Link: <https://www.youtube.com/embed/jxIrPADiMQU>  -Poster Picture: Choose image  -Additional Picture: Choose image  -Film status: Now Showing.  -Film Content: fsgffsdfgddg  4. Click “Save changes” button. | Web application shows error message “Required” under Author fields. | Pass: 4  Fail: 1 | From 21/7/2018  To 5/8/2018 |
| **ACP06** | Edit a Film without Film Genre | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Film” button.  . In List Film view, click “Detail” button.  3. Input fields:  - Film name: Lộ Mặt  - Date: 20/7/2018  -Restricted: 16  -Film Length: 110  -Author: Vĩnh Thuyên  -Film Genre:  -Actor List:  Hứa Minh Đạt, Mạc Văn Khoa.  -Countries: Việt Nam  -Trailer Link: <https://www.youtube.com/embed/jxIrPADiMQU>  -Poster Picture: Choose image  -Additional Picture: Choose image  -Film status: Now Showing.  -Film Content: fsgffsdfgddg  4. Click “Save changes” button. | Web application shows error message “Required” under Film Genre fields. | Pass: 4  Fail: 1 | From 21/7/2018  To 5/8/2018 |
| **ACP07** | Edit a Film without Actor List | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Film” button.  . In List Film view, click “Detail” button.  3. Input fields:  - Film name: Lộ Mặt  - Date: 20/7/2018  -Restricted: 16  -Film Length: 110  -Author: Vĩnh Thuyên  -Film Genre: Hài hước, tình cảm.  -Actor List:  -Countries: Việt Nam  -Trailer Link: <https://www.youtube.com/embed/jxIrPADiMQU>  -Poster Picture: Choose image  -Additional Picture: Choose image  -Film status: Now Showing.  -Film Content: fsgffsdfgddg  4. Click “Save changes” button. | Web application shows error message “Required” under Actor List fields. | Pass: 4  Fail: 1 | From 21/7/2018  To 5/8/2018 |
| *Table 107: Web application test case- <Admin> Edit Film* | | | | | |

* + - 1. <Admin> Add Group Cinema

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **AAGC01** | Add a Group Cinema | 1. Login successfully with role “Admin”.  2. In left sidebar view, click “Manager Group Cinema” button.  2. In List Group Cinema view, click “Create Group Cinema” button.  3. Input fields:  - Group Name: Star Cine  -Address: 524 Quang Trung Gò Vấp  -Phone: 0123456789  -logoImg: StarCine.jpg  4. Click “Save changes” button. | Web application shows message: “Create group cinema successfully”. | Pass: 7  Fail: 3 | From 1/3/2017  To 7/3/2017 |
| *Table 108: Web application test case- <Admin> Add Group Cinema* | | | | | |

* + - 1. <Partner>Add Employee

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **PCE01** | Create an account with role Employee | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manager Employee” button.  2. In All Employee view, click “Add Employee” button.  3. Input fields:  - Employee username: hoang  -Password: abcdef  -Re-confirm Password: abcdef  -Select Cinema: BHD Star Bitexco  -Emloyee Name: Hoàng  -Phone:0123456789  -Email: abc@gmail.com  4. Click “Save Add” button. | Web application shows message: “Create account successfully”. | Pass: 7  Fail: 3 | From 1/3/2017  To 7/3/2017 |
| **PCE02** | Create employee account without username | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manager Employee” button.  2. In All Employee view, click “Add Employee” button.  3. Input fields:  - Employee username:  -Password: abcdef  -Re-confirm Password: abcdef  -Select Cinema: BHD Star Bitexco  -Emloyee Name: Hoàng  -Phone:0123456789  -Email: abc@gmail.com  4. Click “Save Add” button. | Web application shows error message “Required” under username fields. | Pass: 0  Fail: 4 | From 1/3/2017  To 7/3/2017 |
| **PCE03** | Create employee account with user name has existed | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manager Employee” button.  2. In All Employee view, click “Add Employee” button.  3. Input fields:  - Employee username: hoang  -Password: abcdef  -Re-confirm Password: abcdef  -Select Cinema: BHD Star Bitexco  -Emloyee Name: Hoàng  -Phone:0123456789  -Email: abc@gmail.com  4. Click “Save Add” button. | Web application shows error message “username existed”. | Pass: 8  Fail: 2 | From 1/3/2017  To 7/3/2017 |
| **PCE04** | Create employee account without password | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manager Employee” button.  2. In All Employee view, click “Add Employee” button.  3. Input fields:  - Employee username: hoang  -Password: abcdef  -Re-confirm Password: abcdef  -Select Cinema: BHD Star Bitexco  -Emloyee Name: Hoàng  -Phone:0123456789  -Email: abc@gmail.com  4. Click “Save Add” button. | Web application shows error message “Required” under Password fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **PCE04** | Create employee account with re-confirm password incorrect | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manager Employee” button.  2. In All Employee view, click “Add Employee” button.  3. Input fields:  - Employee username: hoang  -Password: abcdef  -Re-confirm Password: abcdefg  -Select Cinema: BHD Star Bitexco  -Emloyee Name: Hoàng  -Phone:0123456789  -Email: abc@gmail.com  4. Click “Save Add” button. | Web application shows error message “Incorrect” under Re-confirm Password fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **PCE05** | Create employee account without Cinema | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manager Employee” button.  2. In All Employee view, click “Add Employee” button.  3. Input fields:  - Employee username: hoang  -Password: abcdef  -Re-confirm Password: abcdef  -Select Cinema:  -Emloyee Name: Hoàng  -Phone:0123456789  -Email: abc@gmail.com  4. Click “Save Add” button. | Web application shows error message “Required” under Select Cinema fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **PCE06** | Create employee account without Phone | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manager Employee” button.  2. In All Employee view, click “Add Employee” button.  3. Input fields:  - Employee username: hoang  -Password: abcdef  -Re-confirm Password: abcdef  -Select Cinema: BHD Star Bitexco  -Emloyee Name: Hoàng  -Phone:  -Email: abc@gmail.com  4. Click “Save Add” button. | Web application shows error message “Required” under Phone fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **PCE07** | Create employee account without Email | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manager Employee” button.  2. In All Employee view, click “Add Employee” button.  3. Input fields:  - Employee username: hoang  -Password: abcdef  -Re-confirm Password: abcdef  -Select Cinema: BHD Star Bitexco  -Emloyee Name: Hoàng  -Phone:0123456789  -Email:  4. Click “Save Add” button. | Web application shows error message “Required” under Email fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| *Table 109: Web application test case- <Partner> Add Employee* | | | | | |

* + - 1. <Partner> Edit Employee

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **PEE01** | Edit an account with role Employee | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manager Employee” button.  2. In All Employee view, click “Detail” button.  3. Input fields:  -Emloyee Name: Hoàng  -Phone:0123456789  -Email: abc@gmail.com  4. Click “Save” button. | Web application shows message: “Edit account successfully”. | Pass: 7  Fail: 3 | From 1/3/2017  To 7/3/2017 |
| **PEE02** | Edit employee account without Employee name | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manager Employee” button.  2. In All Employee view, click “Detail” button.  3. Input fields:  -Emloyee Name:  -Phone:0123456789  -Email: abc@gmail.com  4. Click “Save” button. | Web application shows error message “Required” under Employee name fields. | Pass: 0  Fail: 4 | From 1/3/2017  To 7/3/2017 |
| **PEE03** | Edit employee account without Phone | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manager Employee” button.  2. In All Employee view, click “Detail” button.  3. Input fields:  -Emloyee Name: Hoàng  -Phone:  -Email: abc@gmail.com  4. Click “Save” button. | Web application shows error message “Required” under Phone fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **PEE04** | Edit employee account without Email | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manager Employee” button.  2. In All Employee view, click “Detail” button.  3. Input fields:  -Emloyee Name: Hoàng  -Phone:0123456789  -Email:  4. Click “Save” button. | Web application shows error message “Required” under Email fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| *Table 110: Web application test case- <Partner> Edit Employee* | | | | | |

* + - 1. <Partner> Delete Employee

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **ACP01** | Delete an account with role Employee | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manager Employee” button.  3. In List Partner Account view, click “Delete” button.  4. Click button “Yes” when asked for confimation | Web application shows the account disappear from the list. | Pass: 7  Fail: 3 | From 21/7/2018  To 5/8/2018 |
| *Table 111: Web application test case- <Partner> Delete Employee* | | | | | |

* + - 1. <Partner> Add Cinema

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **PCC01** | Create a Cinema | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manage Cinema” button.  2. In All Employee view, click “Add Cinema” button.  3. Input fields:  - Cinema name: Star Quang Trung  -Address: abcdef  -Phone:0123456789  -Email: [abc@gmail.com](mailto:abc@gmail.com)  -Open Time: 8:00-23:00  -Introduction: afsssfdfdsfds  -Choose file: Choose cinema image  4. Click “Add” button. | Web application shows message: “Create Cinema successfully”. | Pass: 7  Fail: 3 | From 1/3/2017  To 7/3/2017 |
| **PCC02** | Create a Cinema without Cinema name | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manage Cinema” button.  2. In All Employee view, click “Add Cinema” button.  3. Input fields:  - Cinema name:  -Address: abcdef  -Phone:0123456789  -Email: [abc@gmail.com](mailto:abc@gmail.com)  -Open Time: 8:00-23:00  -Introduction: afsssfdfdsfds  -Choose file: Choose cinema image  4. Click “Add” button. | Web application shows error message “Required” under Cinema name fields. | Pass: 0  Fail: 4 | From 1/3/2017  To 7/3/2017 |
| **PCC03** | Create a Cinema with Cinema name has existed | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manage Cinema” button.  2. In All Employee view, click “Add Cinema” button.  3. Input fields:  - Cinema name: Star Quang Trung  -Address: abcdef  -Phone:0123456789  -Email: [abc@gmail.com](mailto:abc@gmail.com)  -Open Time: 8:00-23:00  -Introduction: afsssfdfdsfds  -Choose file: Choose cinema image  4. Click “Add” button.  . | Web application shows error message “Cinema name existed”. | Pass: 8  Fail: 2 | From 1/3/2017  To 7/3/2017 |
| **PCC04** | Create a Cinema without Address | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manage Cinema” button.  2. In All Employee view, click “Add Cinema” button.  3. Input fields:  - Cinema name: Star Quang Trung  -Address:  -Phone:0123456789  -Email: [abc@gmail.com](mailto:abc@gmail.com)  -Open Time: 8:00-23:00  -Introduction: afsssfdfdsfds  4. Click “Add” button. | Web application shows error message “Required” under Address fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **PCC05** | Create a Cinema without Phone | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manage Cinema” button.  2. In All Employee view, click “Add Cinema” button.  3. Input fields:  - Cinema name: Star Quang Trung  -Address: abcdef  -Phone:  -Email: [abc@gmail.com](mailto:abc@gmail.com)  -Open Time: 8:00-23:00  -Introduction: afsssfdfdsfds  -Choose file: Choose cinema image  4. Click “Add” button. | Web application shows error message “Required” under Select Phone fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **PCC06** | Create a Cinema without Email | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manage Cinema” button.  2. In All Employee view, click “Add Cinema” button.  3. Input fields:  - Cinema name: Star Quang Trung  -Address: abcdef  -Phone:0123456789  -Email:  -Open Time: 8:00-23:00  -Introduction: afsssfdfdsfds  -Choose file: Choose cinema image  4. Click “Add” button. | Web application shows error message “Required” under Select Email fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| *Table 112: Web application test case- <Partner> Add Cinema* | | | | | |

* + - 1. <Partner> Edit Cinema

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **PCC01** | Edit a Cinema | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manage Cinema” button.  2. In All Cinema view, click “Detail” button.  3. Input fields:  - Cinema name: Star Quang Trung  -Address: abcdef  -Phone:0123456789  -Email: [abc@gmail.com](mailto:abc@gmail.com)  -Open Time: 8:00-23:00  -Introduction: afsssfdfdsfds  -Choose file: Choose cinema image  4. Click “Save” button. | Web application shows message: “Edit Cinema successfully”. | Pass: 7  Fail: 3 | From 1/3/2017  To 7/3/2017 |
| **PCC02** | Edit a Cinema without Cinema name | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manage Cinema” button.  2. In All Cinema view, click “Detail” button.  3. Input fields:  - Cinema name:  -Address: abcdef  -Phone:0123456789  -Email: [abc@gmail.com](mailto:abc@gmail.com)  -Open Time: 8:00-23:00  -Introduction: afsssfdfdsfds  -Choose file: Choose cinema image  4. Click “Save” button. | Web application shows error message “Required” under Cinema name fields. | Pass: 0  Fail: 4 | From 1/3/2017  To 7/3/2017 |
| **PCC03** | Edit a Cinema with Cinema name has existed | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manage Cinema” button.  2. In All Cinema view, click “Detail” button.  3. Input fields:  - Cinema name: Star Quang Trung  -Address: abcdef  -Phone:0123456789  -Email: [abc@gmail.com](mailto:abc@gmail.com)  -Open Time: 8:00-23:00  -Introduction: afsssfdfdsfds  -Choose file: Choose cinema image  4. Click “Save” button. | Web application shows error message “Cinema name existed”. | Pass: 8  Fail: 2 | From 1/3/2017  To 7/3/2017 |
| **PCC04** | Edit a Cinema without Address | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manage Cinema” button.  2. In All Cinema view, click “Detail” button.  3. Input fields:  - Cinema name: Star Quang Trung  -Address:  -Phone:0123456789  -Email: [abc@gmail.com](mailto:abc@gmail.com)  -Open Time: 8:00-23:00  -Introduction: afsssfdfdsfds  4. Click “Save” button. | Web application shows error message “Required” under Address fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **PCC05** | Edit a Cinema without Phone | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manage Cinema” button.  2. In All Cinema view, click “Detail” button.  3. Input fields:  - Cinema name: Star Quang Trung  -Address: abcdef  -Phone:  -Email: [abc@gmail.com](mailto:abc@gmail.com)  -Open Time: 8:00-23:00  -Introduction: afsssfdfdsfds  -Choose file: Choose cinema image  4. Click “Add” button. | Web application shows error message “Required” under Select Phone fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| **PCC06** | Edit a Cinema without Email | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manage Cinema” button.  2. In All Cinema view, click “Detail” button.  3. Input fields:  - Cinema name: Star Quang Trung  -Address: abcdef  -Phone:0123456789  -Email:  -Open Time: 8:00-23:00  -Introduction: afsssfdfdsfds  -Choose file: Choose cinema image  4. Click “Save” button. | Web application shows error message “Required” under Select Email fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| *Table 113: Web application test case- <Partner> Edit Cinema* | | | | | |

* + - 1. <Partner> Add Room

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **PAR01** | Add a room | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manage Cinema” button.  2. In All Cinema view, click “Detail” button.  3. Click Add Room button in Edit Cinema View  4. Input fields:  - Room name: Rạp 1  -Number of seats: 120  5. Click “Suggest” button.  6. Use elements from the right side menu to rrange the seats of the room.  7. When all seats are arranged click “Save” button.  8. Click “Save” button in Edit Cinema View. | Web application shows message: “Add Room successfully”. | Pass: 7  Fail: 3 | From 1/3/2017  To 7/3/2017 |
| **PAR02** | Click “Save” button right after the Add Room View appeared. | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manage Cinema” button.  2. In All Cinema view, click “Detail” button.  3. Click Add Room button in Edit Cinema View  4. Click “Save” button. | Web application shows error message “Required”. | Pass: 0  Fail: 4 | From 1/3/2017  To 7/3/2017 |
| **PAR03** | Click “Suggest” button right after the Add Room View appeared. | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manage Cinema” button.  2. In All Cinema view, click “Detail” button.  3. Click Add Room button in Edit Cinema View  4. Click “Suggest” button. | Web application shows error message “Required” under Room name fields. | Pass: 8  Fail: 2 | From 1/3/2017  To 7/3/2017 |
| **PAR04** | Click “Suggest” button when the input of filed” Number of seats” is lower than 80 or greater than 330. | 1. Login successfully with role “Partner”.  2. In left sidebar view, click “Manage Cinema” button.  2. In All Cinema view, click “Detail” button.  3. Click Add Room button in Edit Cinema View  4. Input fields:  - Room name: Rạp 1  -Number of seats: lower than 80 or greater than 330  5. Click “Suggest” button. | Web application shows error message “Number of seats must be greater or equal to 80 and lower or equal to 330” under Number of seats fields. | Pass: 4  Fail: 1 | From 1/3/2017  To 7/3/2017 |
| *Table 114: Web application test case- <Partner> Add Room* | | | | | |

* + 1. Mobile Application Test Cases
       1. <User> Login

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **MUL01** | Login | 1.Go to Login View  2. Fill in user name and password and click “Login” or login using gmail, facebook. | Mobile application shows user login successfully with the profile of the accunt logged in. | Pass: 8  Fail: 2 | From 5/7/2018  To 15/7/2018 |
| **MUL02** | Login with incorrect username or password | 1.Go to Login View  2. Fill in user name and password incorrectly | Mobile application shows error message” Incorrect username or password. | Pass: 8  Fail: 2 | From 5/7/2018  To 15/7/2018 |
| Table 115: *Mobile application test case- <User> login* | | | | | |

* + - 1. <User> View List Film

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **MUL01** | View List of Film that are currently showing | 1.Go to Home Page  2. Click on the tab “Đang Chiếu”. | Mobile application shows a list of film that are currently showing. | Pass: 8  Fail: 2 | From 5/7/2018  To 15/7/2018 |
| Table 116: *Mobile application test case- <User> View List Film* | | | | | |

* + - 1. <Authorized User>Purchase Ticket

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test case procedure** | **Expected output** | **Result** | **Test Date** |
| **MAUBT01** | Booking a ticket | 1. Login successfully  2. Go to home page  3. Go to List of Film That are currently showing View.  4. Choose a movie  5. Choose date to watch the film  6. Choose group of cinema.  7. Cho time of the date to watch the film.  8. Choose type of seat and number of seat.  9. click “Tiếp tục”  10. Choose seat based on the seat map of the cinema room  11. click “Tiếp tục”  12. input fields:  - email: “gravelysin44@gmail.com”  - phone: “0908853025”  13. click button “Đặt vé” | Mobile application shows info of the ticket | Pass: 8  Fail: 2 | From 5/7/2018  To 15/7/2018 |
| **MAUBT02** | Booking a ticket with blank or incorrect email format | 1. Login successfully  2. Go to home page  3. Go to List of Film That are currently showing View.  4. Choose a movie  5. Choose date to watch the film  6. Choose group of cinema.  7. Cho time of the date to watch the film.  8. Choose type of seat and number of seat.  9. click “Tiếp tục”  10. Choose seat based on the seat map of the cinema room  11. click “Tiếp tục”  12. input fields:  - email: “”  - phone: “0908853025”  13. click button “Đặt vé” | Mobile application shows notification “Email không hợp lệ” . | Pass: 8  Fail: 2 | From 5/7/2018  To 15/7/2018 |
| **MAUBT03** | Booking a ticket with blank or incorrect phone format | 1. Login successfully  2. Go to home page  3. Go to List of Film That are currently showing View.  4. Choose a movie  5. Choose date to watch the film  6. Choose group of cinema.  7. Cho time of the date to watch the film.  8. Choose type of seat and number of seat.  9. click “Tiếp tục”  10. Choose seat based on the seat map of the cinema room  11. click “Tiếp tục”  12. input fields:  - email: “”  - phone: “0908853025”  13. click button “Đặt vé” | Web application shows notification “Số điện thoại không hợp lệ”. | Pass: 8  Fail: 2 | From 5/7/2018  To 15/7/2018 |
| **MAUBT04** | Choose seat position with a blank seat at the start or end of a row of seats | 1. Login successfully  2. Go to home page  3. Go to List of Film That are currently showing View.  4. Choose a movie  5. Choose date to watch the film  6. Choose group of cinema.  7. Cho time of the date to watch the film.  8. Choose type of seat.  9. Choose 2 seats.  10. click “Tiếp tục”  11. Choose seat B02, B03  12. click “Tiếp tục” | Mobile application show notification “Vui long không để trống vị trí đầu dãy hoặc cuối dãy” | Pass: 9  Fail: 1 | From 5/7/2018  To 15/7/2018 |
| **MAUBT05** | Choose seat position with a blank seat at the start or end of a row of seats | 1. Login successfully  2. Go to home page  3. Go to List of Film That are currently showing View.  4. Choose a movie  5. Choose date to watch the film  6. Choose group of cinema.  7. Cho time of the date to watch the film.  8. Choose type of seat.  9. Choose 2 seats.  10. click “Tiếp tục”  11. Choose seat B01, B03  12. click “Tiếp tục” | Mobile application show notification “Vui long không để trống vị trí ở giữa 2 ghế chọn” | Pass: 9  Fail: 1 | From 5/7/2018  To 15/7/2018 |
| Table 117: *Mobile application test case- <User> Purchase Ticket* | | | | | |

1. Software User’s Manual
   1. Installation Guide
      1. Setting up environment at server side

The specifications are based on the dependencies requirements and performance test result from previous section of this document.

* + - 1. Hardware Requirements

|  |  |
| --- | --- |
| **Hardware** | **Specification** |
| Internet Connection | 4 Mbps |
| Computer Processor | Intel® Core i3 1.4GHz or more |
| Computer Memory | 3GB of RAM or more |
| Hard Drive | 50GB or more |
| *Table 118: Installation Guide - Hardware Requirements* | |

* + - 1. Software Requirements

|  |  |
| --- | --- |
| **Software** | **Application name/version** |
| Operating System | Windows |
| DotNet Core | .NET Core 2.1 |
| DBMS | SQL Server |
| *Table 119: Installation Guide - Software Requirement* | |

* + - 1. Mobile Requirements

|  |  |
| --- | --- |
| **Hardware** | **Specification** |
| CPU | Quad-core 1500 MHz, Krait |
| Memory | 2 GB |
| Storage | 16 GB |
| Model | Google Pixel |
| *Table 120: Installation Guide - Mobile Requirement* | |

* 1. User Guide
     1. Web application
        1. Resell ticket

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Picture | Step |  |
| 1 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\ban_lai_ve_8.png | 1 | In home page, click link “Bán lại vé”. |
| ban_lai_ve_9 | 2 | Enter your email to textbox “Email”. |
| 3 | Click button “Tìm kiếm” to search all ticket buyed. |
| 2 | ban_lai_ve_10 | 4 | Check your email to get code confirm. |
| 3 | ban_lai_ve_3 | 5 | Enter code in mail to textbox. |
| 6 | Click button “Xác nhận”. |
| 4 | ban_lai_ve_12 | 7 | Enter description for reason you resell ticket. |
| 8 | Click button “Đăng bán”. |
| 5 | ban_lai_ve_11 | 9 | Enter email buyer. |
| 10 | Click button “Xác nhận bán”. |
| Table 121: User Guide - Web application – Resell ticket | | | |

* + - 1. Booking ticket

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Picture | Step |  |
| 1 | muave_0 | 1 | Choice and click schedule of film. |
| 2 | muave1 | 2 | Choice number ticket you want to buy. |
| 3 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\muave2.png | 3 | Click button “Chọn Ghế” to redirect to screen choice seat. |
| 4 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\muave3.png |  | In screen choice seat, you can choice your seats. |
| 5 | muave4 | 5 | Enter your email. |
| 6 | Enter your phone number. |
| 7 | Choice payment method. |
| 6 | muave6 | 8 | Click button “Đặt vé” to start payment. |
| 7 |  | 9 | Payment |
| Table 122: User Guide - Web application –Booking ticket | | | |

* + 1. Mobile application
       1. Login

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Picture | Step |  |
| 1 | dangnhap_1 | 1 | In screen “Cá Nhân”, table “TÀI KHOẢN” you can touch picture to start login. |
| 2 | dang nhap | 2 | In screen login, you enter your username. |
| 3 | In screen login, you enter your password. |
| 4 | In screen login, you touch button “Đăng nhập” to login. |
| Table 123: User Guide - Mobile application – Login | | | |

* + - 1. Register

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Picture | Step |  |
| 1 | dangki_1 | 1 | In screen login, you can touch button “ĐĂNG KÍ” to start register. |
| 2 | Dang ki | 2 | In screen register, you enter your username.  Username must have at least 6 characters. |
| 3 | In screen register, you enter your password.  Password must have at least 8 characters. |
| 4 | In screen register, you enter your email.  Email address must be in the correct format. For example: abc@gmail.com |
| 5 | In screen register, you enter your phone number.  Phone number must be 9-11 character. |
| 6 | Touch button “ĐĂNG KÍ” |
| Table 124: User Guide - Mobile application – Register | | | |

* + - 1. Booking ticket

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Picture | Step |  |
| 1 | film information 1 | 1 | In screen “Khám Phá”, selecting film you want to see. |
| 2 | filmschedule_1 | 2 | In screen schedule film, selecting day you want to see schedule of this film. |
| 3 | In screen schedule film, selecting show time you want and touch this. |
| 3 | android_muave_3 | 4 | In screen Select number ticker, pick number ticket you want to buy. Max ticket is 10. |
| 4 | android_muave_5 | 4 | In screen choice seat, selecting position of seat you want to buy and click “TIẾP TỤC” to continue. |
| 5 | android_muave_4 | 5 | In screen choice seat, touch button “TIẾP TỤC” to continue. |
| 6 | android_muave_6 | 6 | In screen confirm order, you enter your email.  Email address must be in the correct format. |
| 7 | You enter your phone number.  Phone number must be 9-11 character. |
| 7 | android_muave_7 | 8 | In screen confirm order, you choice and touch payment method. |
| 8 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\android_muave_8.png | 9 | In screen confirm order, you touch button “TIÊP TỤC” to continue. |
| Table 125: User Guide - Mobile application – Booking ticket | | | |

* + - 1. Show all order

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Picture | Step |  |
| 1 | vedamua_1 | 1 | In screen User, touch tab “VÉ ĐÃ MUA” to see all ticket you were bought. |
| 2 | vedamua_2 | 2 | After performing the first step, you will see the screen showing all the tickets you have purchased. |
| Table 126: User Guide - Mobile application – Show all order | | | |

* + - 1. Order detail

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Picture | Step |  |
| 1 | ve da mua | 1 | In screen User, selecting order you want to see detail of this order. |
| 2 | user_3 | 2 | After performing the first step, you will see the screen showing detail of order you choice before. |
| Table 127: User Guide - Mobile application – Order detail | | | |

* + - 1. Resell ticket

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Picture | Step |  |
| 1 | banlaive_1 | 1 | In screen Order detail, selecting a ticket you want to resell and touch button “BÁN LẠI VÉ”. |
| 2 | banlaive_2 | 2 | Touch “ĐỒNG Ý” in dialog confirm to continue resell ticket. Or you can touch “HỦY” to cancel resell ticket. |
| 3 | banlaive_3 | 3 | After performing the above steps, your ticket status will become "Đang bán lại". Next step, touch button "XÁC NHẬN BÁN" to continue. |
|  | banlaive_4 | 4 | In dialog confirm email buyer, enter the email of buyer. |
| 5 | Touch “ĐỒNG Ý” to finish resell ticket. |
| Table 128: User Guide - Mobile application – Resell ticket | | | |

* + - 1. Change ticket

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Picture | Step |  |
| 1 | doive_1 | 1 | In screen Order detail, selecting a ticket you want to change and touch button “ĐỔI VÉ”. |
| 2 | doive_2 | 2 | Touch “ĐỒNG Ý” in dialog confirm to continue change ticket or “HỦY” to cancel change ticket. |
| 3 | doive_3 | 3 | In screen choice schedule, you select schedule you want change. |
| 4 | doive_4 | 4 | In screen choice seat, selecting seat you want and click “TIẾP TỤC” to finish change ticket. |
| Table 129: User Guide - Mobile application – Change ticket | | | |

* + 1. Admin web application
       1. Crawl film data and add film to system

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Picture | Step |  |
| 1 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\ca_01.png | 1 | In screen Crawl film data, selecting month you want to crawl data film.  Month selected must be greater than or equal current month in the system. |
| 2 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\ca2.png | 2 | Selecting a film you want add and click button “add film” of this film. You can select multiple films. |
| 3 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\ca4.png | 3 | Click “View Added List” to show add film you selected. |
| 4 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\ca7.png | 4 | If you do not want to add this movie anymore you can click the "remove movie" button to remove this movie from the list. |
| 5 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\ca6.png | 5 | Click button “Save all” to save all films you selected to system. |
| Table 130: User Guide – Admin web application – Crawl film data and add film to system | | | |

* + - 1. Add new film

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Picture | Step |  |
| 1 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\createfilm1.png | 1 | In screen Manager film, click button “Create Film” to start create film. |
| 2 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\cf1.png | 2 | Enter film name. Film name cannot be empty. |
| 3 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\cf3.png | 3 | Enter author and countries. Note: Not require. |
|  | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\cf2.png | 4 | Select film restricted to age limit for watching film. |
| 4 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\cf4.png | 5 | Selecting picture for this film. |
| 5 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\cf5.png | 6 | Enter content for this film. |
| 6 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\cf_3.png | 7 | Select date release for this film. |
| 7 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\cf_5.png | 8 | Enter film length and selecting film status.  There are two states for the film: "Now showing" is for the film being show, "Coming soon" is for the upcoming film. |
| 8 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\cf6.png | 9 | Click “Save changes” to save this film to system. Or “Close” for cancel save film. |
| Table 131: User Guide – Admin web application – Add new film | | | |

* + 1. Manager web application
       1. Basic add new schedule

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Picture | Step |  |
| 1 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\add_0.png | 1 | In screen Basic add schedule, selecting film to add. |
| 2 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\add_1.png | 2 | Selecting room of this cinema to add want to add this schedule. |
| 3 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\add_2.png | 3 | Then, selecting show time. |
| 4 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\add_3.png | 4 | Then, selecting date show. Date must be greater than the current date in the system. |
| 5 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\add_4.png | 5 | Click button “Save” to finish add schedule |
| Table 132: User Guide – Manager web application – Basic add new schedule | | | |

* + - 1. Advantage Add Schedule

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Picture | Step |  |
| 1 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\ad_0.png | 1 | In screen Advantage Add Schedule, selecting date you want manager schedule. Date must be greater than the current date in the system. By default, the system will automatically suggest a movie schedule. |
| 2 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\ad_1.png | 2 | Selecting a film you want add schedule. In this step you can click button “Unsuggest” to remove all default schedule suggest by the system. |
| 3 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\ad_3.png | 3 | Click to show time to change schedule for this film. By default, the system will not let at a time and in a room show two movies. |
| 4 | C:\Users\LacLuu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\ad_4.png | 4 | After changing the schedule, you can click "Save" to finish the schedule. |
| Table 133: User Guide – Manager web application – Advantage Add Schedule | | | |

1. Appendix
2. UML standard diagrams:

<https://www.tutorialspoint.com/uml/uml_standard_diagrams.htm>

1. Component diagram notation:

<https://www.ibm.com/developerworks/rational/library/dec04/bell/>

1. algorithm:

1. UML Diagrams:

<http://www.uml-diagrams.org/>