

## 1. Requirements

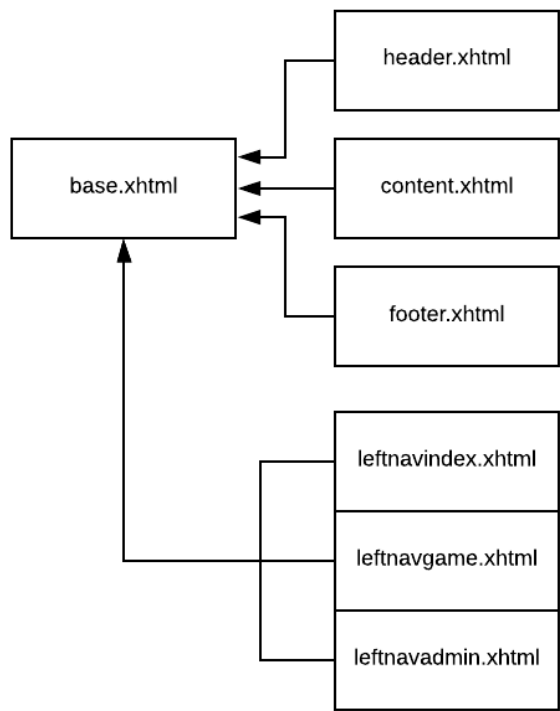
- F1. Any user can view an encyclopedia of unit and building information without logging in.
- F2. Admin can suspend and unsuspend user accounts. Suspended players can log in, but cannot perform any actions and are notified that they are suspended.
- F3. Admin can view, update, and delete user accounts.
- F4. User can create an account with their own username, kingdom name, race, password, email. Email is used for security to send notifications to the player when sensitive actions are taken such as changing password or soft deleting their account.
- F5. User can change their own password – need to enter current password to confirm.
- F6. User can soft delete their account which suspends it. Only admin can unsuspend their account.
- F7. Fiefdoms generate resources and can be attacked even though the user is not logged in
- F8. An event log is generated of all the things that have happened since the player last checked their main page. This is displayed to the user when they check their main page.
- F9. Player can explore to gain new land in order to build new buildings
- F10. Player can spend gold and land to build different buildings that have different effects. For example, building towers helps them when they are attacked by another player. Building mines increases gold income per turn. Building farms increases peasant growth per turn.
- F11. Player can spend gold and peasants to recruit different army units.
- F12. Player can send diplomatic messages to other players, declare war, request peace, request alliance, and break alliances. Allies cannot declare war on each other. Players can only attack each other if they are at war.
- F13. Players can reply to diplomatic messages, or accept/decline diplomatic requests (peace request, alliance requests).
- F14. Player can send a chosen number of units from their army to attack an enemy player they are at war with. This generates a battle report which shows a log of the battle, who won, resources captured etc. If the attacking player wins, they capture some gold, land, and peasants from the defending player. Armies, buildings, resources are updated accordingly for both players.
- F15. Battle reports are persisted and can be viewed at any time, but only by the players who were involved in the battle. Admin can also view battle reports.
- F16. Players can search for and view basic information about all other player kingdoms. E.g. kingdom name, kingdom id, race, diplomatic status, rank, power.
- F17. Each player kingdom has a power rating based on their army size, resources, buildings, land, which is updated every turn. Player rankings are updated accordingly in descending order. Player can view the ranking of all players, as well as their own ranking.
- F18. Users can see how many other unique players are currently logged in
- F19. Admin can create and remove news articles that can be viewed by anyone without having to be logged in. The most recent articles are displayed in a news feed on the index page.

2. Software Design

3 tier architecture.

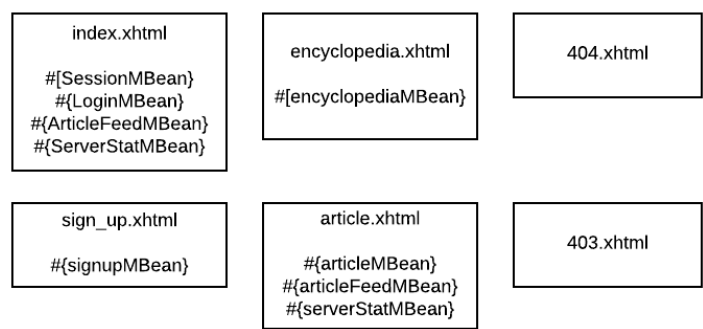
1. Web Layer

JSF Templating



Glassfish resource security

All Rolegroups  
Permitted



## Admin Rolegroup Only

admin.xhtml #{adminMBean}	suspend.xhtml #{suspendUserMBean}	view_user.xhtml #{viewUserMBean}
create_article.xhtml #{createArticleMBean}	update_user.xhtml #{updateUserMBean}	delete_article.xhtml #{deleteArticleMBean}
delete_user.xhtml #{deleteUserMBean}		

## User Rolegroup

account.xhtml #{accountMBean}	create_message.xhtml #{composeMBean}	rankings.xhtml #{rankingsMBean}
battle_reports.xhtml #{searchBRMBean}	diplomacy.xhtml #{inboxMBean} #{relationsMBean}	recruit_army.xhtml #{recruitMBean}
buildings.xhtml #{buildingsMBean}	explore.xhtml #{buildingsMBean} #{exploreMBean}	report.xhtml #{battleReportMBean}
change_pw.xhtml #{changePassMBean}	message.xhtml #{viewThreadMBean}	search_fiefdom.xhtml #{searchFiefdomMBean}
sign_up.xhtml #{sessionMBean} #{fiefdomInfoMBean}	war.xhtml #{warMBean}	

## User and Suspended Rolegroups

main.xhtml #{accountMBean} #{loginMBean} #{fiefdomInfoMBean} #{serverStatMBean} #{sessionMBean} #{eventLogMBean}
--

# Web Layer

## Backing Beans

<b>@RequestScoped SessionMBean</b>  @EJB GameInfoBeanRemote @EJB ServerStateBeanRemote - String fieldomName - String race - ...other session variables  + init() + recordLogin() + recordLogout() + cleanup() + isLoggedIn : bool + ... other getters and setters	<b>@RequestScoped LoginMBean</b>  @EJB UserinfoManagementBeanRemote @Inject SessionMBean  + login(String username, String password) : String + logout() - sha256Hash(String password)	<b>@RequestScoped ArticleFeedMBean</b>  @EJB NewsBeanRemote - ArrayList<ArticleShortDTO> articleLinks  - @PostConstruct init() + getArticleLinks() : ArrayList<ArticleDTO>	<b>@RequestScoped WarMBean</b>  @EJB ArmyBeanRemote @EJB DiplomacyBeanRemote @EJB WarBeanRemote @Inject FieldomInfoMBean - <validated input form fields>  - @PostConstruct init() + attack() : String + archersChanged(AjaxBehaviourEvent event) + ... other ajax listeners + ... other getters
<b>@RequestScoped ServerStatMBean</b>  @EJB ServerStateBeanRemote @Inject SessionMBean  + getOnlineCount() : int + getTurnCount() : int + serverStatus() : String + getSystemTime() : String	<b>@ConversationScoped RecruitMBean</b>  @EJB ArmyBeanRemote @EJB FieldomInfoMBean @Inject Conversation  - <validated form input fields>  - @PostConstruct init() + recruit() : String + totalArchersCostChanged(AjaxBehaviourEvent event) + ... other ajax listeners + ... getters	<b>@RequestScoped RelationsMBean</b>  @EJB DiplomacyBeanRemote - ArrayList<RelationDTO> relations  - @PostConstruct init() + getWars() : ArrayList<RelationDTO> + getAlliances() : ArrayList<RelationDTO> - filterRelations(String relation) : ArrayList<RelationDTO>	<b>@ConversationScoped SearchBRMBean</b>  @EJB WarBeanRemote @Inject Conversation - int id - ArrayList<BattleReportShortDTO> reports  - @PostConstruct init() + viewReport() : String + getReports() : ArrayList<BattleReportShortDTO> + ... other getters
<b>@RequestScoped ReportMBean</b>  @EJB WarBeanRemote - BattleReportDTO br  - @PostConstruct init() + getAttacker() : String + getDefender() : String + ...other getters	<b>@ConversationScoped SearchFieldomMBean</b>  @EJB FieldomBeanRemote @EJB DiplomacyBeanRemote @Inject Conversation - ArrayList<FieldomShortDTO> fieldoms  - @PostConstruct init() + chooseFieldomById() + chooseFieldomByName() - populate(FieldomBasicDTO f) + isExistingFieldom(...params) + ...other getters	<b>@RequestScoped AccountMBean</b>  @EJB UserinfoManagementRemote @Inject SessionMBean @Inject LoginMBean  @ValidPassword, @NotBlank, @Size(min=8) - String delPass  + delete() : String + getPassword() : String + setPassword(String password) + isAdmin() : bool + isUser() : bool + isUser() : bool + isSuspended() : bool	<b>@ConversationScoped ViewThreadMBean</b>  @EJB DiplomacyBeanRemote @EJB FieldomBeanRemote @Inject SessionMBean @Inject Conversation - ThreadDTO thread - ArrayList<MessageDTO> messages  - @PostConstruct init() + reply() : String + accept() : String + decline() : String + delete() : String + getThread() : ThreadDTO + ...other getters and setters
<b>@RequestScoped ChangePassMBean</b>  @EJB UserinfoManagementRemote @Inject LoginMBean  - <validated form input fields>  + changePassword() : String - sha256Hash(String password) : String + validatePasswordPair(...params) + ... getters	<b>@RequestScoped WarMBean</b>  @EJB EventsBeanRemote - ArrayList<TurnDTO> logs  - @PostConstruct init() + getLogs() : ArrayList<TurnDTO>	<b>@ConversationScoped BuildingsMBean</b>  @EJB BuildingsBeanRemote @Inject FieldomInfoMBean @Inject Conversation  - <validated form input fields>  - @PostConstruct init() + build() : String + totalFarmsCostChanged(AjaxBehaviourEvent event) + ... other ajax listeners + ... getters	<b>@ConversationScoped ExploreMBean</b>  @EJB ExploreBeanRemote @Inject Conversation @Inject FieldomInfoMBean  - <validated form input fields>  - @PostConstruct init() + explore() : String + totalCostChanged(AjaxBehaviourEvent event) + ...getters
<b>@RequestScoped FieldomInfoMBean</b>  @EJB FieldomBeanRemote @EJB UserinfoManagementBeanRemote @Inject SessionMBean  - FieldomDTO fDto  - @PostConstruct init() + isSuspended() : bool + ... other getters	<b>@RequestScoped ArticleMBean</b>  @EJB NewsBeanRemote - ArticleFullDTO article  - @PostConstruct init() + getTitle() : String + getBody() : String + getPosted() : String	<b>@RequestScoped ComposeMBean</b>  @EJB DiplomacyBeanRemote @EJB FieldomBeanRemote @Inject SessionMBean  - <validated form input fields>  + send() : String + isValidFieldomId(...params) + isValidType(...params) + ...getters	<b>@RequestScoped InboxMBean</b>  @EJB DiplomacyBeanRemote  - ArayList<ThreadDTO> threads  - @PostConstruct init() + getThreads() : ArrayList<ThreadDTO>

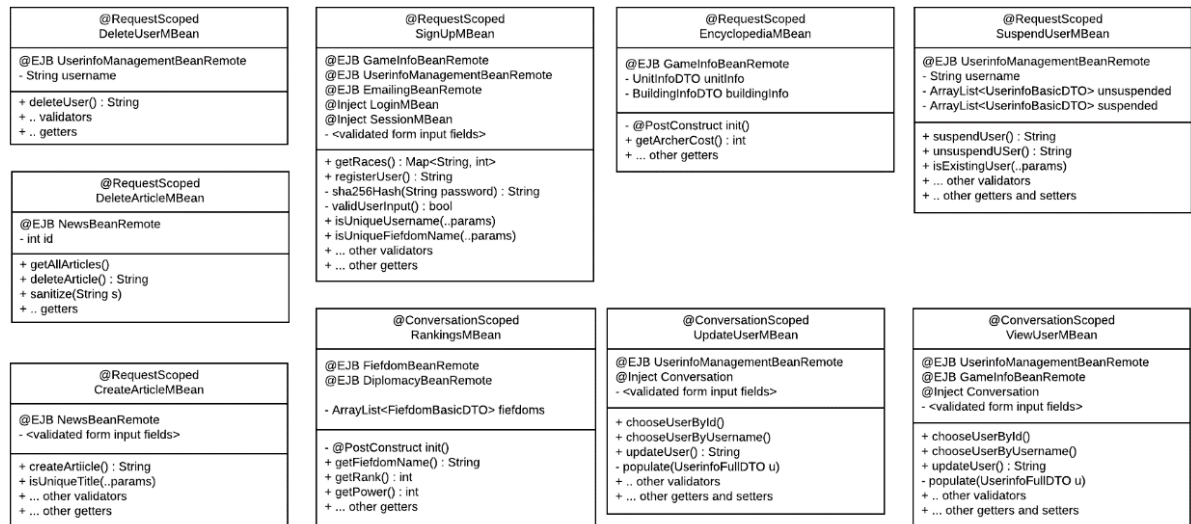


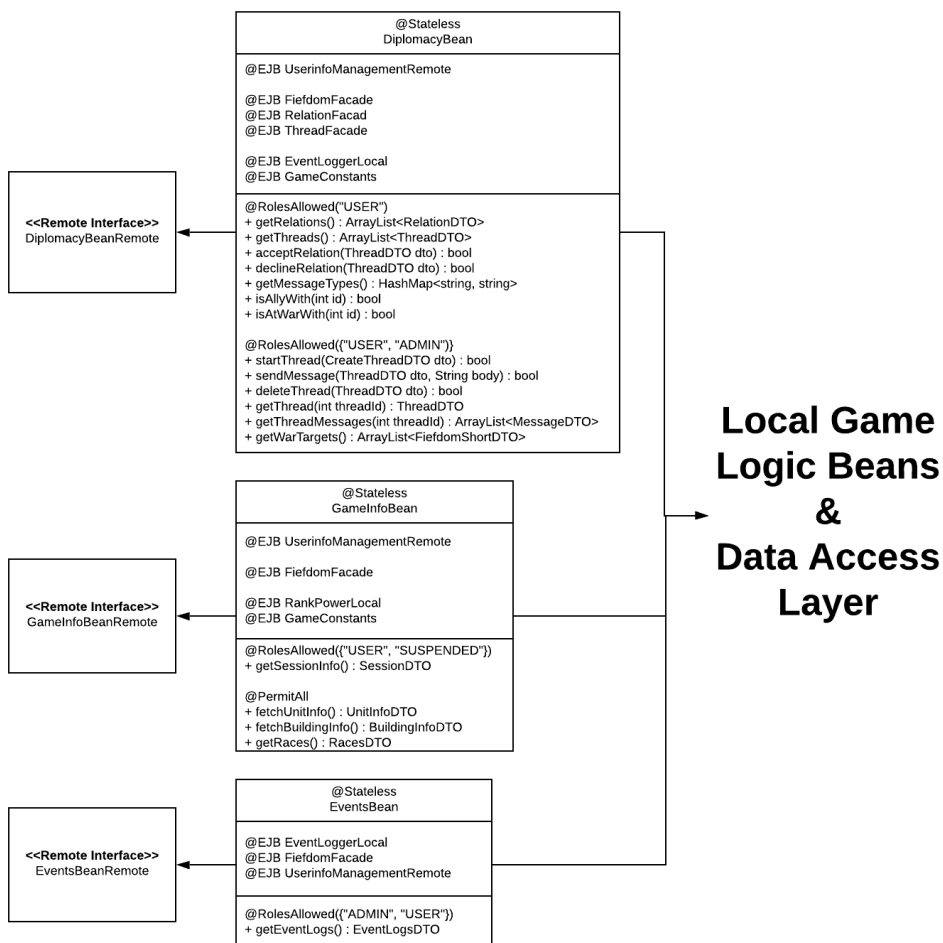
Figure 1.5 – Backing bean support for JSF views following SRP.

## 2. Business Layer

Two sub layers - front facing remote interfaces, and internal EJB's with local interfaces.

Stateless EJB's where possible to optimise scalability. Singletons used to cache non-critical data and avoid database hits. Concurrency via Java EE concurrency management.

## Business Layer Remote Services



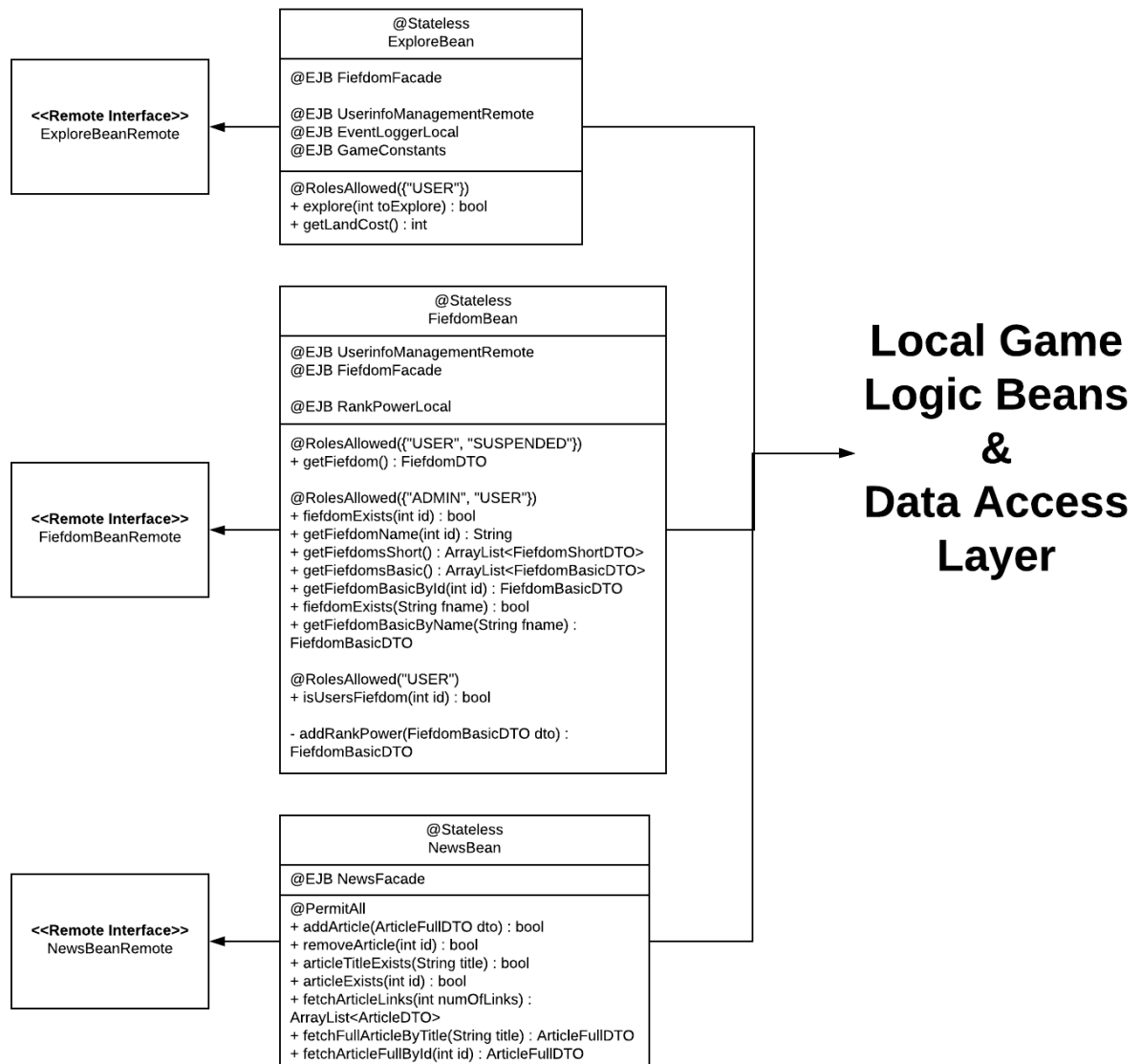


Figure 2.2a – Front facing business layer EJB's

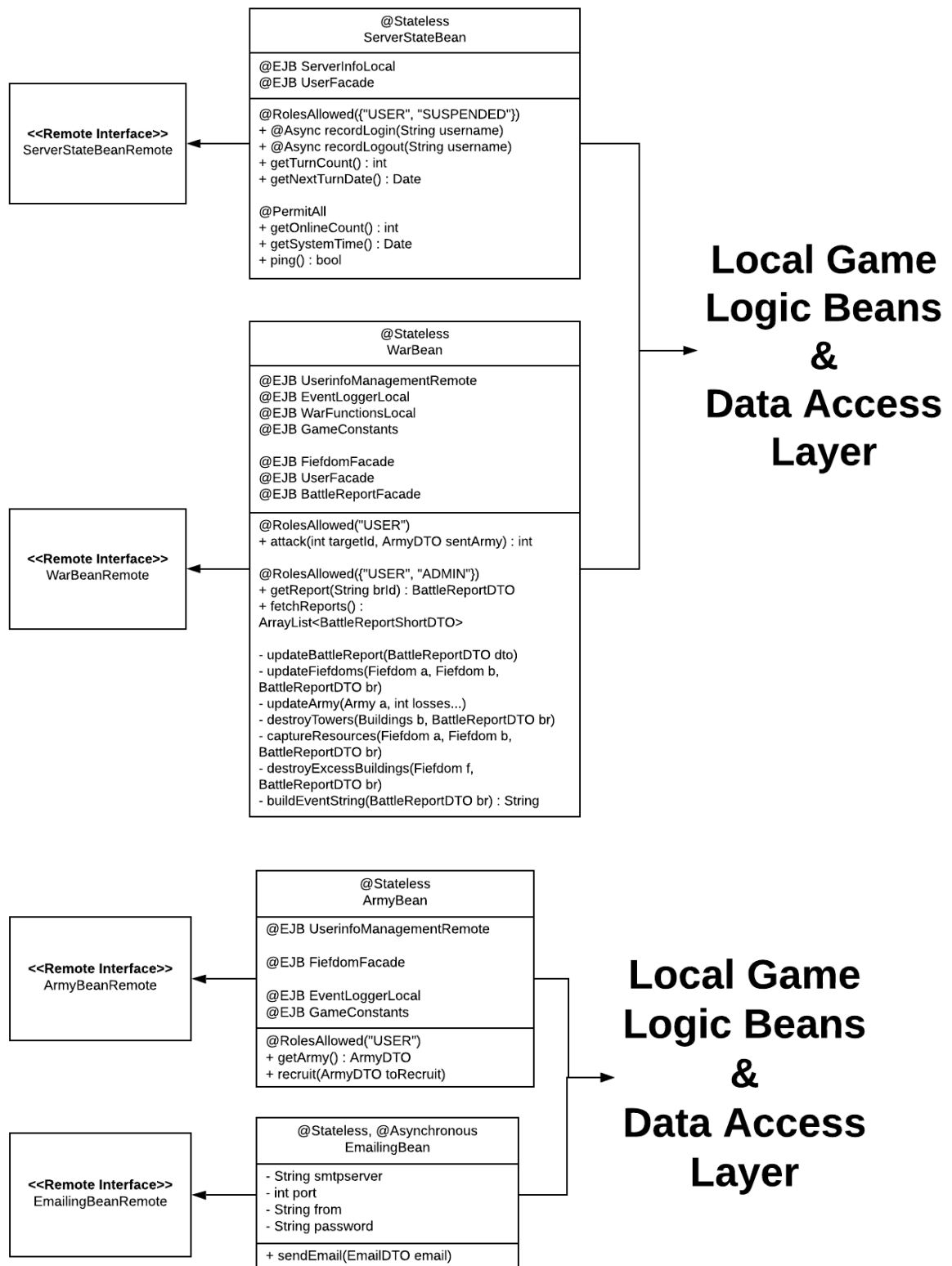


Figure 2.2b – Front facing business layer EJB's

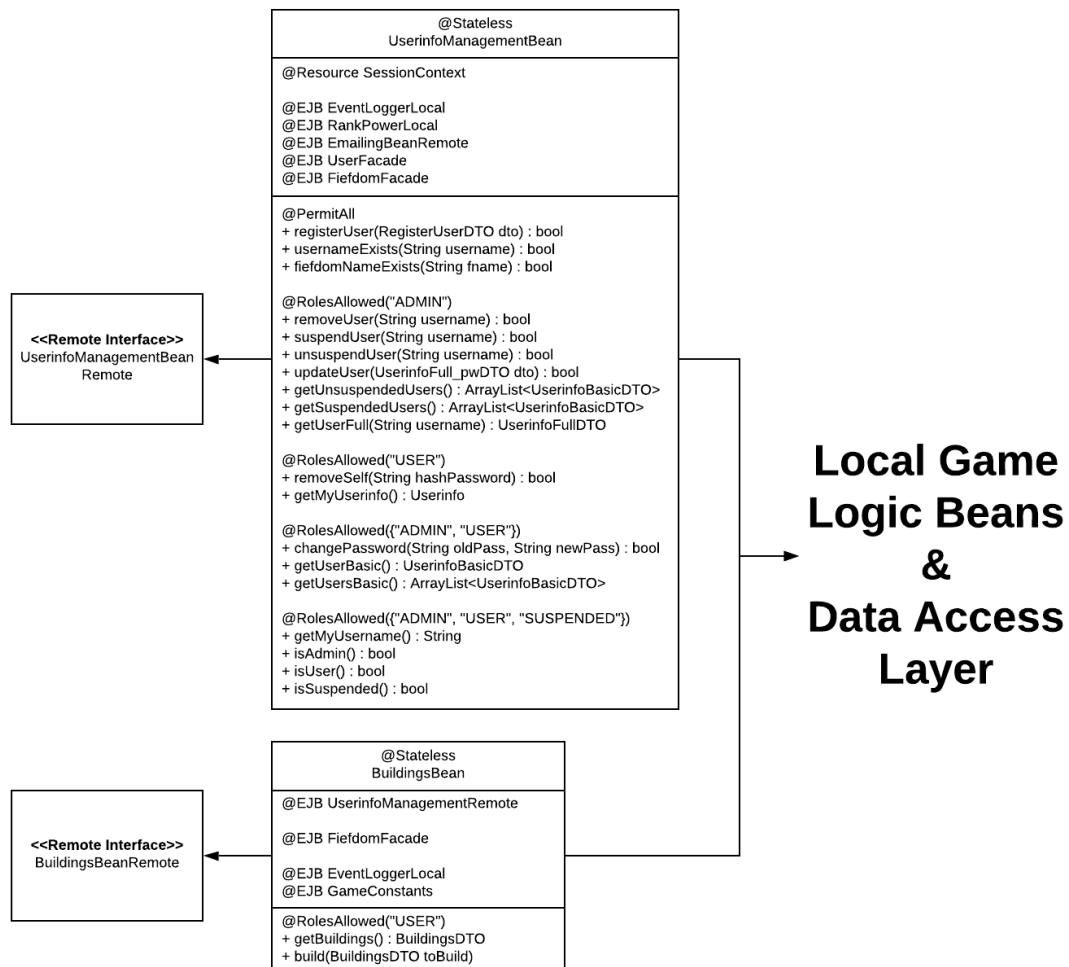
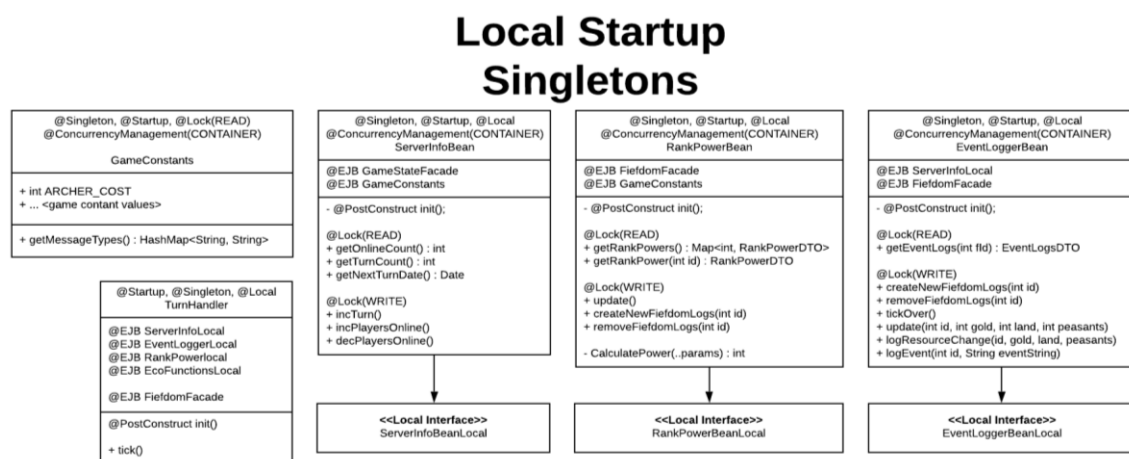
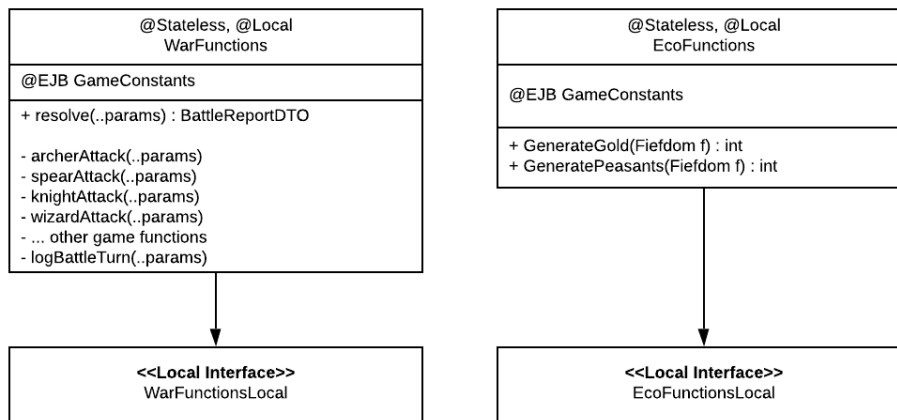


Figure 2.2c – Front facing Business Layer EJB's





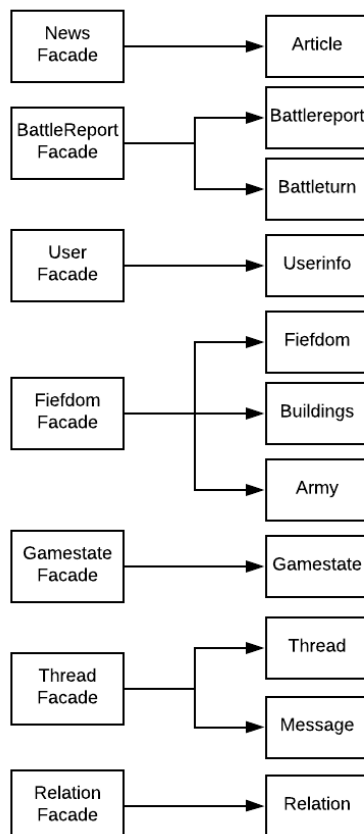
# Local Game Logic Beans



## 3. Data Access Layer

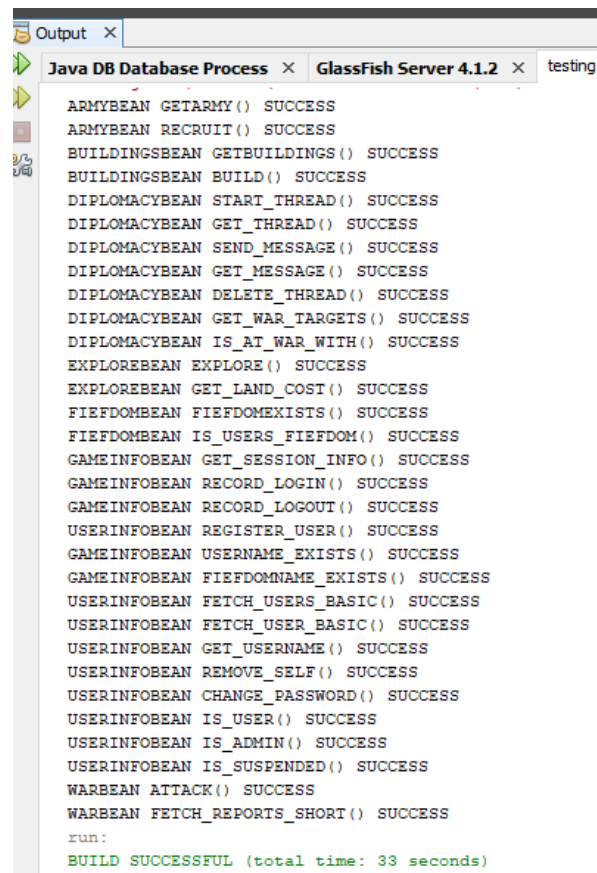
Object persistence via hibernate ORM / JPA via entity facades.

## Data Access Entities



#### 4. Software Testing Results

Testing harness over EJB remote interfaces.



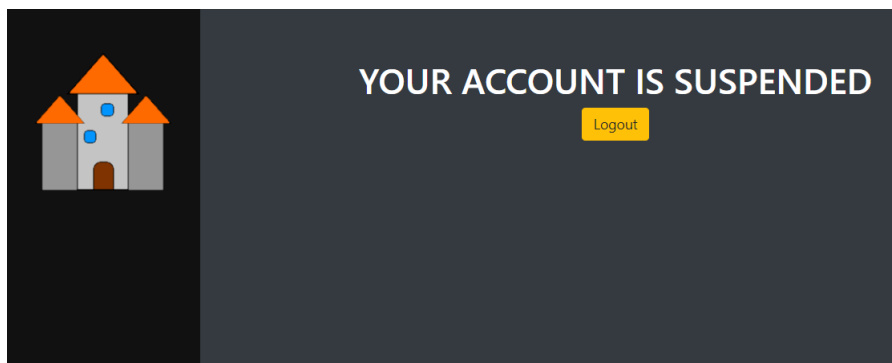
```

Output x
Java DB Database Process x GlassFish Server 4.1.2 x testing
ARMYBEAN GETARMY() SUCCESS
ARMYBEAN RECRUIT() SUCCESS
BUILDINGSBEAN GETBUILDINGS() SUCCESS
BUILDINGSBEAN BUILD() SUCCESS
DIPLOMACYBEAN START_THREAD() SUCCESS
DIPLOMACYBEAN GET_THREAD() SUCCESS
DIPLOMACYBEAN SEND_MESSAGE() SUCCESS
DIPLOMACYBEAN GET_MESSAGE() SUCCESS
DIPLOMACYBEAN DELETE_THREAD() SUCCESS
DIPLOMACYBEAN GET_WAR_TARGETS() SUCCESS
DIPLOMACYBEAN IS_AT_WAR_WITH() SUCCESS
EXPLOREBEAN EXPLORE() SUCCESS
EXPLOREBEAN GET_LAND_COST() SUCCESS
FIEFDOMBEAN FIEFDOMEXISTS() SUCCESS
FIEFDOMBEAN IS_USERS_FIEFDOM() SUCCESS
GAMEINFOBEAN GET_SESSION_INFO() SUCCESS
GAMEINFOBEAN RECORD_LOGIN() SUCCESS
GAMEINFOBEAN RECORD_LOGOUT() SUCCESS
USERINFOBEAN REGISTER_USER() SUCCESS
GAMEINFOBEAN USERNAME_EXISTS() SUCCESS
GAMEINFOBEAN FIEFDOMNAME_EXISTS() SUCCESS
USERINFOBEAN FETCH_USERS_BASIC() SUCCESS
USERINFOBEAN FETCH_USER_BASIC() SUCCESS
USERINFOBEAN GET_USERNAME() SUCCESS
USERINFOBEAN REMOVE_SELF() SUCCESS
USERINFOBEAN CHANGE_PASSWORD() SUCCESS
USERINFOBEAN IS_USER() SUCCESS
USERINFOBEAN IS_ADMIN() SUCCESS
USERINFOBEAN IS_SUSPENDED() SUCCESS
WARBEAN ATTACK() SUCCESS
WARBEAN FETCH_REPORTS_SHORT() SUCCESS
run:
BUILD SUCCESSFUL (total time: 33 seconds)
  
```

Manual testing of JSF views and templating

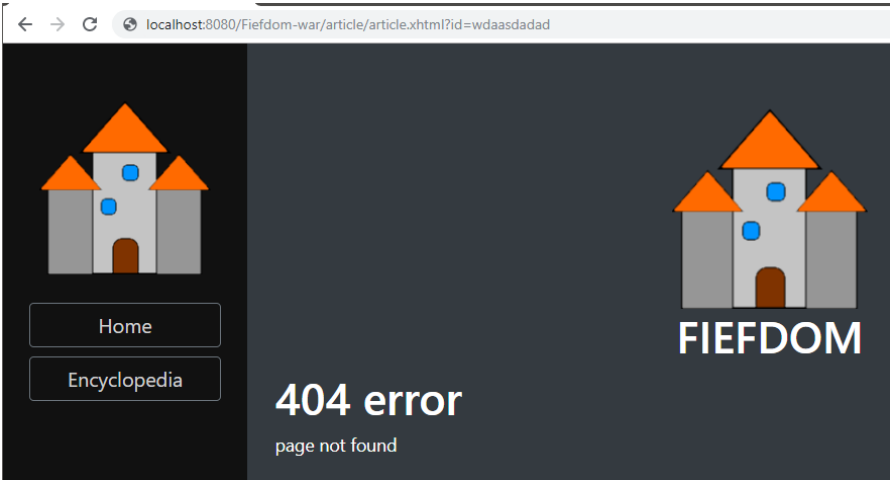
#### Testing Suspended user

1. Log in as Admin, suspend user, logout
2. Log in as suspended user

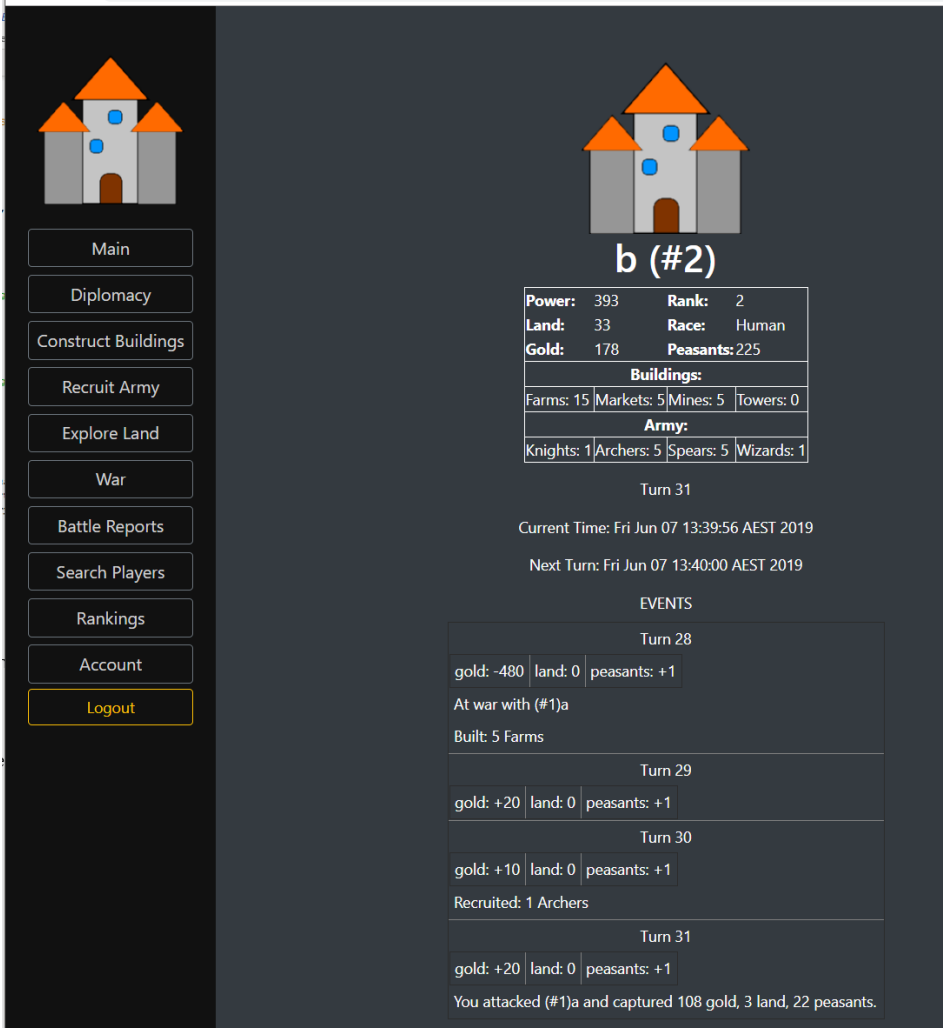


## Test Invalid Requests


Invalid id for battle reports and article get requests



## Test Event Logging



Test AJAX listeners



Main

Diplomacy

Construct Buildings

Recruit Army

Explore Land

War


Battle Reports

Search Players

Rankings

Account

Logout



FIEFDOM

Recruitment

Gold: 2234

Peasants: 261

Required Gold: 300

Required Peasants: 30

Unit	Recruited	Unit Cost	Amount	Total Cost
Archers	0	10	<input type="text" value="10"/>	100
Spearmen	0	10	<input type="text" value="20"/>	200
Knights	0	50	<input type="text" value="0"/>	0
Wizards	0	100	<input type="text" value="0"/>	0

Recruit

Test JSF formatting



Main

Diplomacy

Construct Buildings

Recruit Army

Explore Land

War

Battle Reports

Search Players

Rankings

Account

Logout

Battle Report (#1)

Date: Fri Jun 07 13:39:50 AEST 2019

Attacker (#2)b wins

	Attacker	Defender
	(#2)b	(#1)a
Army	Archers: 6 Spears: 5 Knights: 1 Wizards: 1	Archers: 5 Spears: 5 Knights: 1 Wizards: 1
Losses	Archers: 1 Spears: 0 Knights: 0 Wizards: 0	Archers: 5 Spears: 5 Knights: 1 Wizards: 1

Captured Resources:  
Gold: 108 Land: 3 Peasants: 22

Battle Log

Turn	Attacker	Defender
1	6 Archers attack ->	4 Archers killed
2	Units damaged <-	1 Archers attack
3	5 Spears attack ->	Units damaged
4	Units damaged <-	5 Spears attack
5	1 Knights attack ->	Units damaged
6	1 Archers killed <-	1 Knights attack
7	1 Wizards attack ->	1 Archers,2 Spears,1 Wizards killed
8	5 Archers attack ->	1 Spears killed
9	5 Spears attack ->	Units damaged

## Test Web Resource Security Constraints

Accessing restricted resources without appropriate authentication

