

Empires of Avatharia

Group 22

Christopher Raschke

Johan Wessman

Jashar Ghavampour

Johan Granberg

5.1 CRC Cards

Class GameEngine

Contains all the algorithms that are needed for the game to work. In a sense this class *is* the game

Collaborators:

All in-game classes
(see 5.2)
Startup
HTML Generator
DB

Class Startup

Contains all the startup values that are needed for a new “game session” to be created. This class accesses predefined startup variables and creates necessary instances of different in-game classes.
Note that these “sessions” are often many weeks or even months long.

Collaborators:

All in-game classes (see 5.2)
GameEngine

Class HTML Generator

Generates the HTML pages that eventually end up at the client. It handles the requests that are sent from the HTML server and forwarded to it via the GameEngine. It contains information about how to get pictures etc from the database as well as HTML generating methods.

Collaborators:

All in-game classes
(see 5.2)
DB

Class EventQueue

Acts like a normal queue, but for all the events in the system. When an action is performed it is added to the queue and processed in order by the GameEngine.

Collaborators:

GameEngine

Class Province

A player owns provinces.
Each province contains Buildings.
A province is placed in an area, (specific coordinates)
A province contains two armies

Collaborators:

Player
Building
Area
Army

Class Area

An area is basically square coordinates of the world-map. An area can contain a province owned by a player.

Collaborators:

Province

Class Building

The building class contains properties about a building and what units and upgrades can be bought if a player owns that kind of building.
A province contains buildings.

Collaborators:

None

Class Upgrade

Collaborators:

An upgrade contains some specific property for either a unit or a building that is gained when it is bought.

An upgrade also contains a cost.

Class Unit

The class's main use is to store unit specific data.

Store unit upgrades

Class Player

Store information about the players armies.

Store information about the players Provinces

Store the player's faction.

Store other player specific information.

Class Army

Collect units and group them into an army

Store information about the army's current mission.

Class Sides

Collect armies attacking and defending the same target and sort them into two sides

Class Faction

Store information regarding class features.

Store information regarding class game stats.

Class Combat

Calculate combat results

Report combat results to CombatLog

Class Combat

Generate a combatlog for the players to read.

Class User

Handles login procedure, sessions, user permissions and user information.

Class Db

Handles database connection

Unit

Building

Collaborators:

Army

Upgrades

Collaborators:

Faction

Province

Collaborators:

Unit

Player

Sides

Collaborators:

Army

Combat

Collaborators:

Player

Collaborators:

Sides

CombatLog

Collaborators:

Combat

Collaborators:

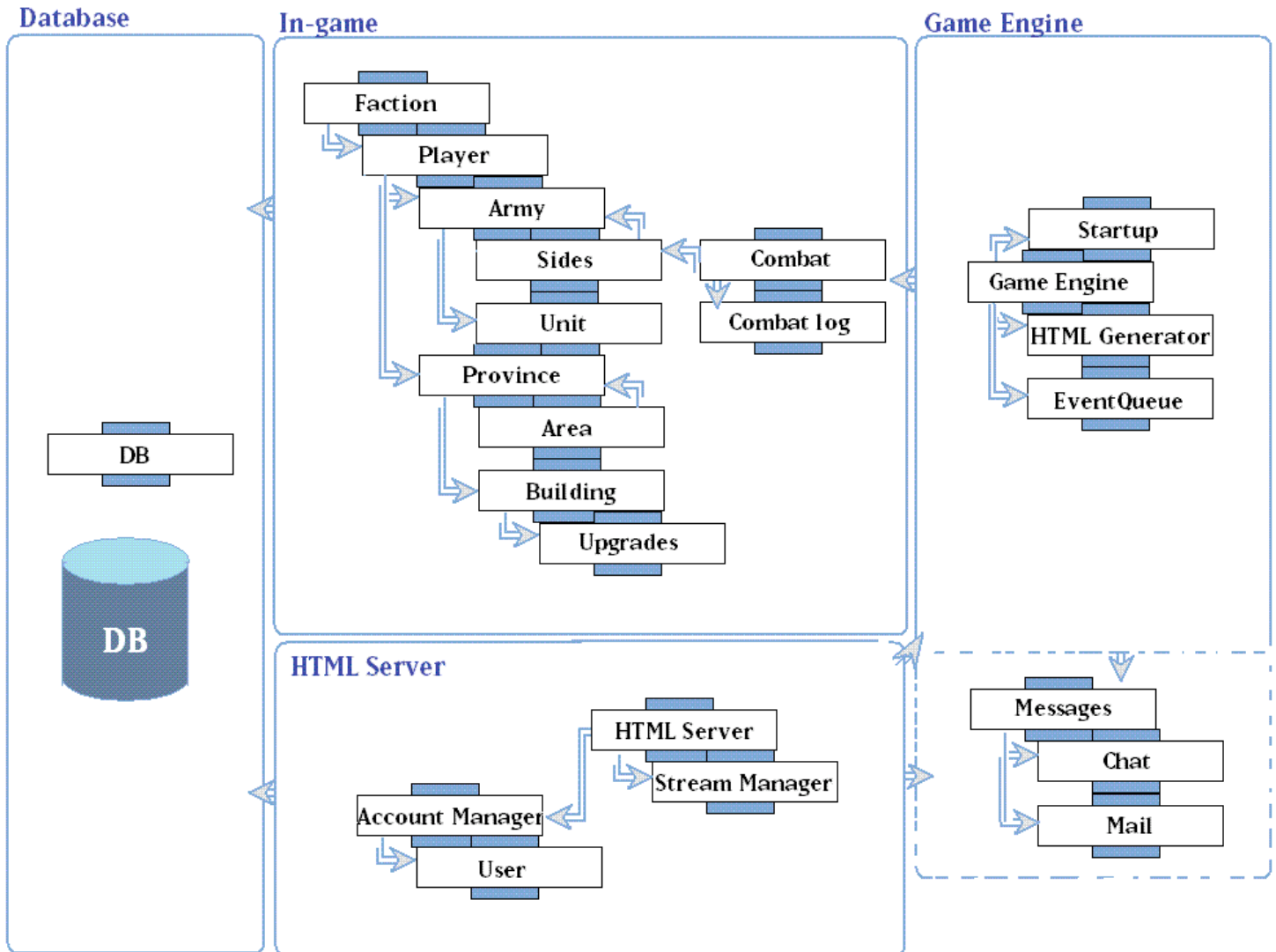
Player

Db

Collaborators:

User

5.2 Class Diagram



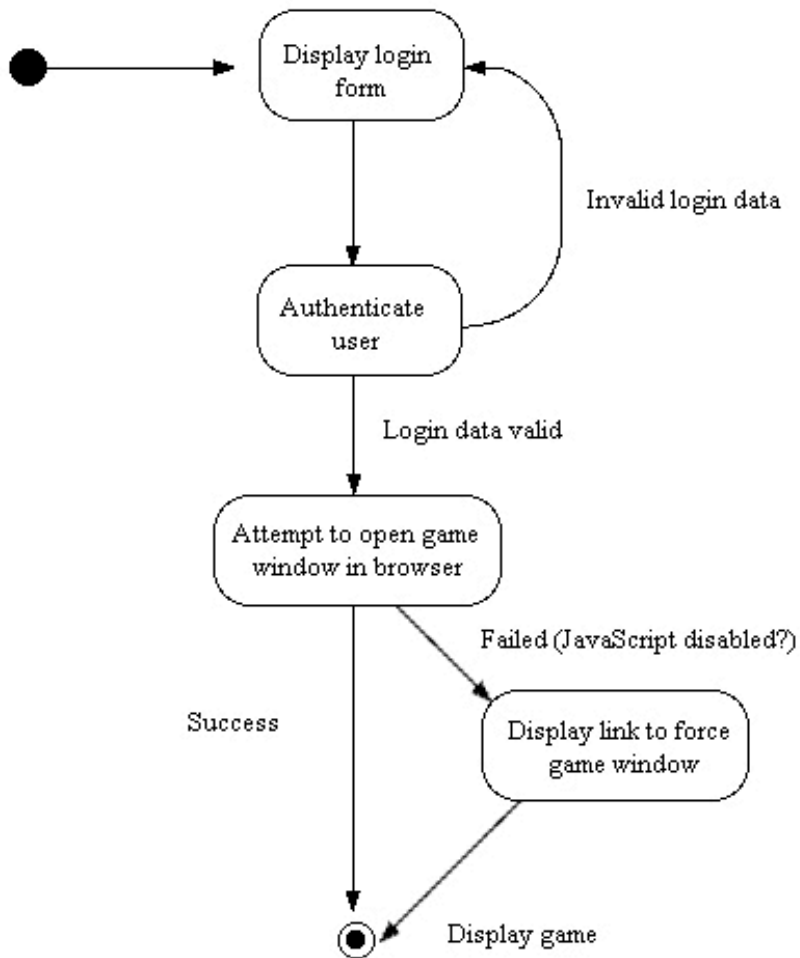
The class diagram should be interpreted as:

- Each box with a name on it is a class
- An arrow between classes basically means “contains”. For example a “faction” contains a lot of “players” and a “player” has “armies” but the when the player is “in combat” the “sides” also contains “armies”.
- Boxes surrounding classes’ means that they belong to the same group. Classes in the same group are strongly connected and are more likely to appear in the same “folder”/”level of architecture” on the server.
- The arrow between the “boxes” indicates how requests are sent between the different groups. These arrows do represent arrows from all classes in the group. The reverse of these arrows could be interpreted as responses or data transfers.

Example: all classes in the In-game group need to be able to send requests to the database. If the Game Engine wants to generate a HTML page, via the HTML generator, containing “the province view” it will send a request to the province to send pictures of all its buildings. The province will then send this request forward to all its buildings who will then ask the “DB” class for the pictures from the database.

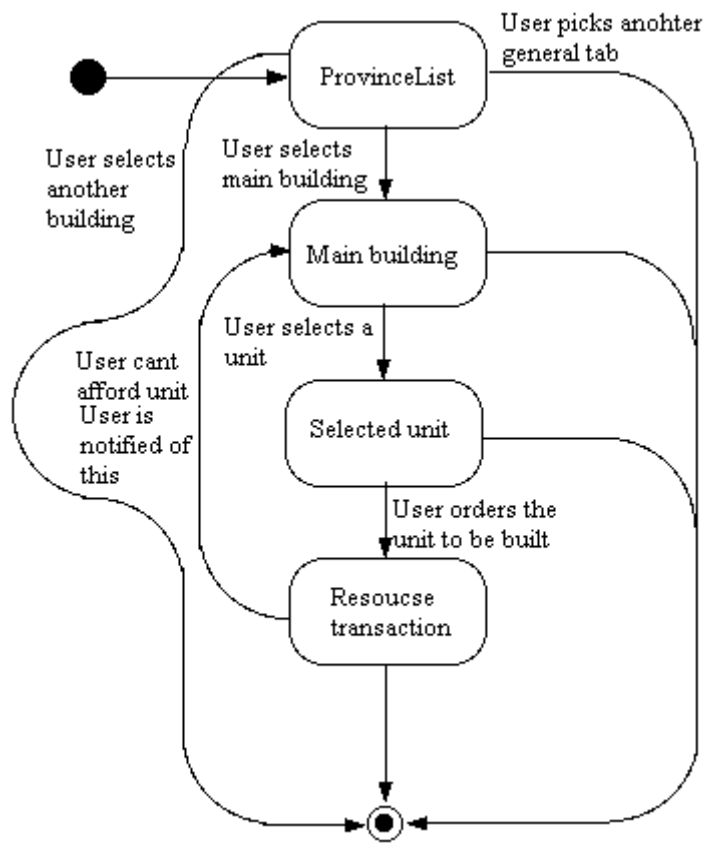
5.3 State charts

State Chart, Logging in



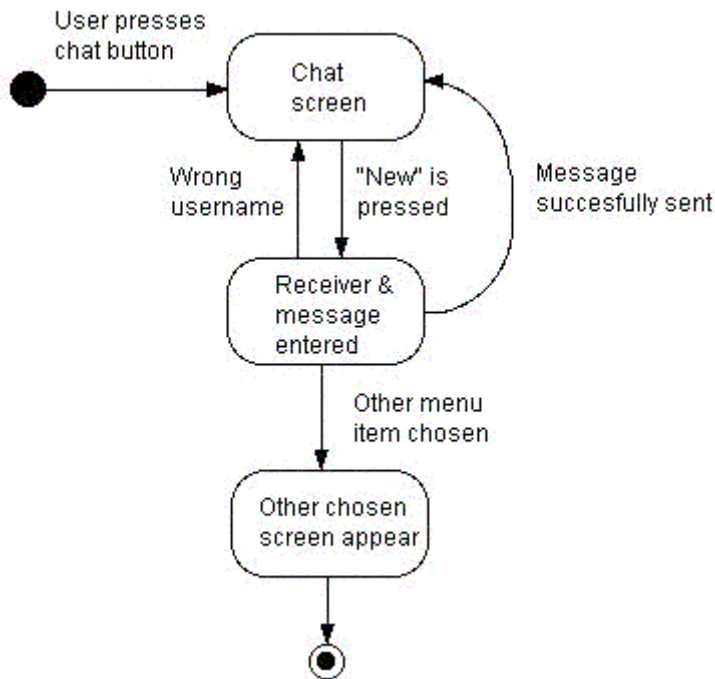
State Chart, Training a unit

It is assumed that the player is logged in into the game.



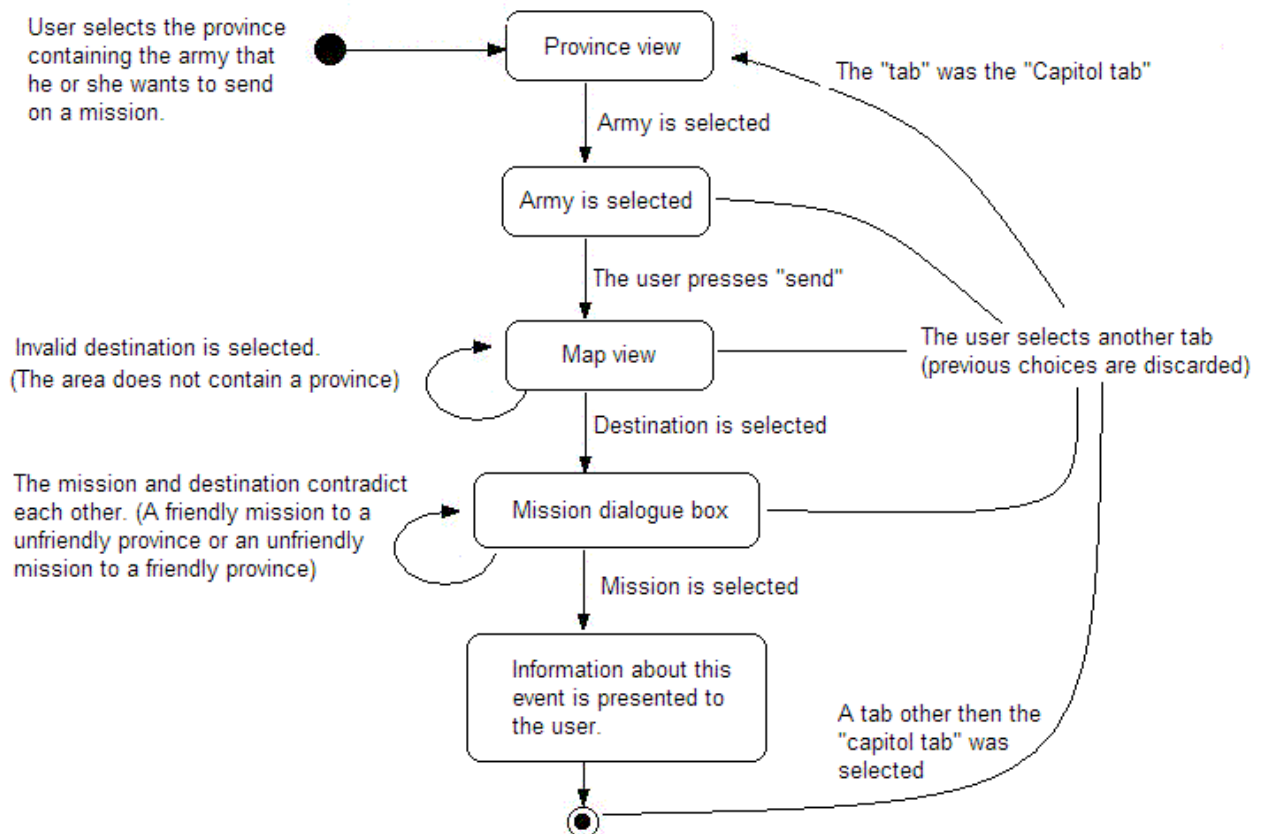
State Chart, Sending Mail

It is assumed that the player is logged in into the game.



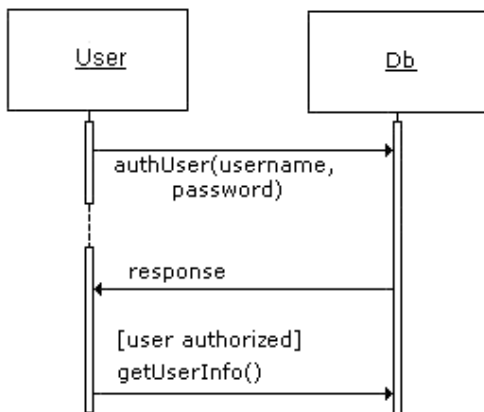
State Chart, Sending an army to a mission

It is assumed that the player is logged in into the game.



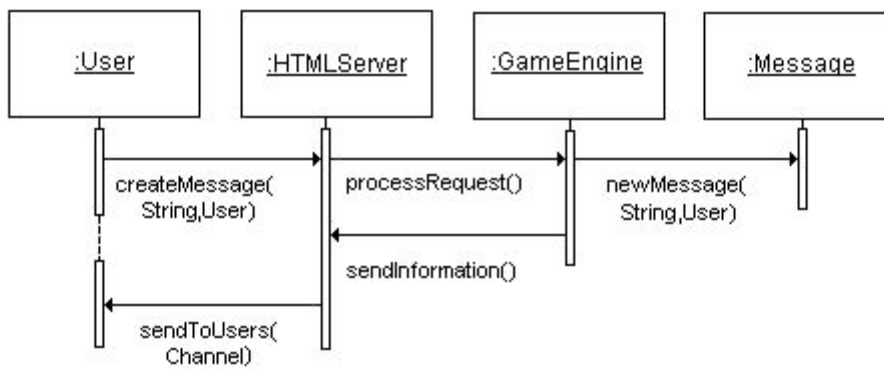
5.4 Interaction charts

Interaction Chart, Logging in



Interaction Chart, Instant Chat.

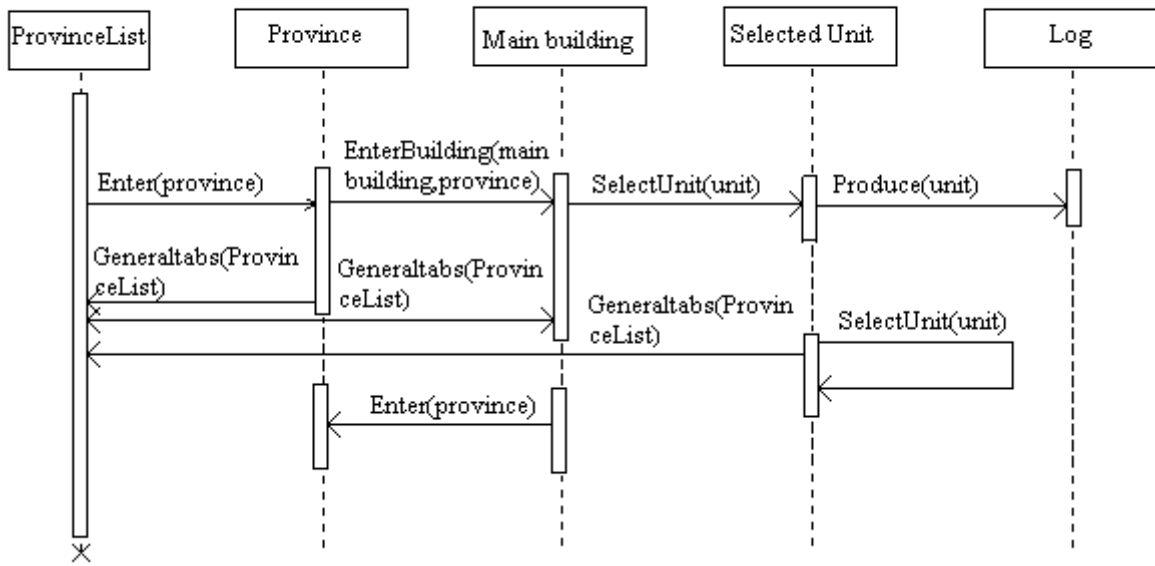
It is assumed that the player is logged in into the game.



Interaction Chart, Train unit

It is assumed that the player is logged in into the game.

Produce a Unit sequence diagram



Interaction Chart, Build Building

It is assumed that the player is logged in into the game.

Produce a building sequence diagram

