SE 307 PROJECT REPORT

Umut Aydın

Pelin TUNÇ

Sami Kutay Erdoğan

Commits on Dec 19, 2023

Display reworked, Color added for Properties, reworked Chance and Community Tiles, Starting Tile Fixed

skutayerdogan committed 18 hours ago

d29d022

Commits on Dec 18, 2023

Player order fix

umutaydins committed 2 days ago

f8ea628

fix for cards

umutaydins committed 2 days ago

59c06e3

little fix

umutaydins committed 2 days ago

4d19e25

Jail tekrar düzenlendi commentler yapıldı

pelintuncc committed 2 days ago

441ccaf

Commits on Dec 17, 2023

Chance , community cards ve baslangic

umutaydins committed 3 days ago

192b014

Player'a renk eklendi Dipslay güncellendi Train ve Utility Tilelar gözden geçirildi

skutayerdogan committed 3 days ago

d27200b

Commits on Dec 16, 2023

Display

skutayerdogan committed 4 days ago

0ca4622

Start,Luxury and Parking Tiles

skutayerdogan committed 4 days ago

72e8eed

Commits on Dec 13, 2023

Ev, otel islemleri, oyunun ana akisi , jail kismi

pelintuncc committed last week

4bbcbbb

Commits on Dec 12. 2023

Commits on Dec 12, 2023

Player new features
umutaydins committed last week
79204a0

Merge branch 'main' of https://github.com/umutaydins/MonopolyProject
umutaydins committed last week
9aed17e

property class'i satin alma islemleri
pelintuncc committed last week
936ef5e

Community and Chance cards
umutaydins committed last week
5334ad6

Train Station
skutayerdogan committed last week
4514679

Kutay
skutayerdogan committed last week
a2be4f8

Commits on Dec 11, 2023

Little fixes
umutaydins committed last week
9a65e18

Commits on Dec 10, 2023

purchase ve jail islemleri
pelintuncc committed last week
823f084

Card classları ve utility class'ı eklendi
pelintuncc committed 2 weeks ago
d10c1cf

Commits on Dec 9, 2023

Tilelar olusturuldu
pelintuncc committed 2 weeks ago
3745260

Commits on Dec 8, 2023

Property class'ı eklendi
pelintuncc committed 2 weeks ago
9bb245e

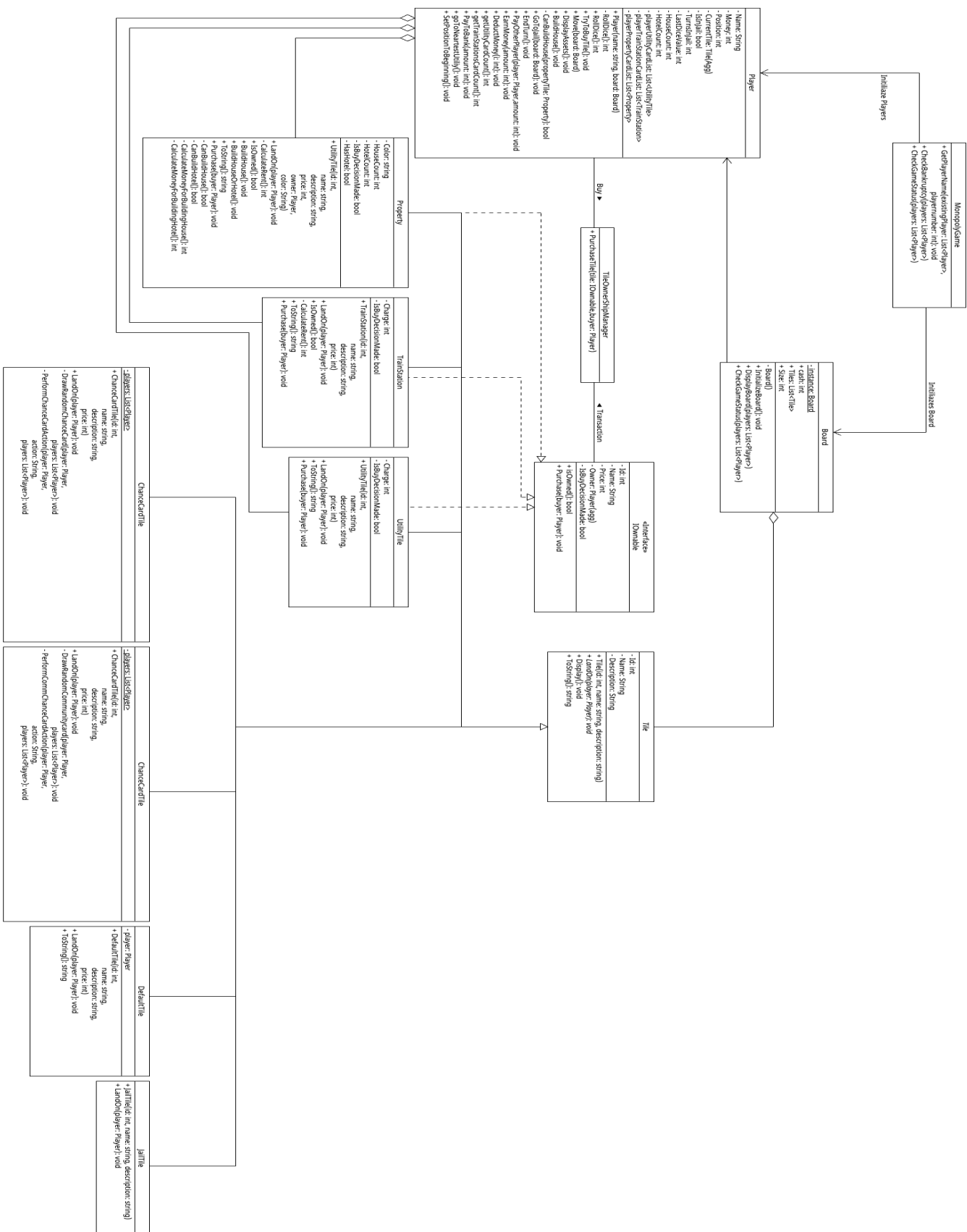Commits on Dec 2, 2023

First push
umutaydins committed 3 weeks ago
4323494

Commits on Nov 24, 2023

First Commit
umutaydins committed last month
7f2ed0d

MonopolyGame
+ GetPlayerName(existingPlayer: List<Player>, playernumber: int): void
+ CheckBankruptcy(players: List<Player>): void
+ CheckGameStatus(players: List<Player>): void

Initialize Players
Initializes Board

Player
- Name: String
- Money: int
- Position: int
- CurrentTile: Tile[Agg]
- IsInJail: bool
- TurnsInJail: int
- LandDiceValue: int
- TurnsOplt: int
- HouseCount: int
- HotelCount: int
- LandDiceValue: int
- BuildHouse(): void
- DisplayAssets(): void
- playerUtilityCard: List<UtilityTile>
- playerTrainstationCard: List<TrainStation>
- playerPropertyCard: List<Property>
+ Player(name: string, board: Board)
+ RollDice(): int
+ TryToBuyTile(): void
+ MoveBoard(Board)
+ BuildHouse(): void
+ CanBuildHouse(property:Tile Property): bool
+ GoToJail(board:Board): void
+ EndTurn(): void
+ PayOtherPlayer(player: Player,amount: int): void
+ EarnMoney(amount: int): void
+ DeductMoney(: int): void
+ getUtilityCardCount(): int
+ getTrainStationsCardCount(): int
+ PayToBank(amount: int): void
+ goToNearestUtility(): void
+ setPosition1beginning(): void

Board
- Instance: Board
+ cash: int
+ Tiles: List<Tile>
- Size: int
- Board()
+ InitializeBoard(): void
+ DisplayBoard(players: List<Player>): void
+ CheckGameStatus(players: List<Player>)

Buy ▶

TileOwnerShipManager
+ PurchaseTile(tile: IOwnable,buyer: Player)

Property
- Color: string
- HouseCount: int
- HotelCount: int
- IsBuyDecisionMade: bool
- HasHotel: bool
+ UtilityTile(id: int,
  name: string,
  description: string,
  price: int,
  owner: Player,
  color: String)
+ LandOn(player: Player): void
+ CalculateRent(): int
+ IsOwned(): bool
+ BuildHouse(): void
+ BuildHouseOrHotel(): void
+ ToString(): string
+ Purchase(buyer: Player): void
+ BuildHouse(): bool
+ CanBuildHouse(): bool
+ CanBuildHotel(): bool
+ CalculateMoneyForBuildingHouse(): void
+ CalculateMoneyForBuildingHotel(): int

TrainStation
- Charge: int
- IsBuyDecisionMade: bool
+ TrainStation(id: int,
  name: string,
  description: string,
  price: int)
+ LandOn(player: Player): void
+ IsOwned(): bool
+ CalculateRent(): int
+ ToString(): string
+ Purchase(buyer: Player): void

UtilityTile
- Charge: int
- IsBuyDecisionMade: bool
+ UtilityTile(id: int,
  name: string,
  description: string,
  price: int)
+ LandOn(player: Player): void
+ ToString(): string
+ Purchase(buyer: Player): void

◀ Transaction

«interface»
IOwnable
- Id: int
- Name: String
- Price: int
- Owner: Player[agg]
- IsBuyDecisionMade: bool
+ IsOwned(): bool
+ Purchase(buyer: Player): void

Tile
- Id: int
- Name: String
- Description: String
+ Tile(id: int, name: string, description: string)
+ LandOn(player: Player): void
+ Display(): void
+ ToString(): string

ChanceCardTile
- players: List<Player>
+ ChanceCardTile(id: int,
  name: string,
  description: string,
  price: int)
+ LandOn(player: Player): void
- DrawRandomChanceCard(player: Player,
  players: List<Player>): void
- PerformChanceCardAction(player: Player,
  action: String,
  players: List<Player>): void

ChanceCardTile
- players: List<Player>
+ ChanceCardTile(id: int,
  name: string,
  description: string,
  price: int)
+ LandOn(player: Player): void
- DrawRandomCommunityCard(player: Player,
  players: List<Player>): void
- PerformCommChanceCardAction(player: Player,
  action: String,
  players: List<Player>): void

DefaultTile
- player: Player
+ DefaultTile(id: int,
  name: string,
  description: string,
  price: int)
+ LandOn(player: Player): void
+ ToString(): string

JailTile
+ JailTile(id: int, name: string, description: string)
+ LandOn(player: Player): void

Class Overview

MonopolyGame:

Responsibility: Controls the flow of the game.

Methods:

Initialize game state.
Get player names.
Check for bankruptcy.
Check game status.

Player:

Responsibility: Represents participants in the game.
Attributes:
Money, position, properties owned (via aggregation with Tile).
Methods:
Roll dice.
Buy tiles.
Move.
Build houses.

Board:

Responsibility: Represents the game board.
Methods:
Initialize the board.
Display the board.
Check the game status.

Tile:

Responsibility: Abstract class for different spaces on the board.
Attributes:
ID, name, description.
Methods:
Actions when a player lands.
Display tile information.

IOwnable:

Responsibility: Interface for objects that can be owned.
Methods:
isOwned.
Purchase.

TileOwnershipManager:

Responsibility: Manages ownership of IOwnable tiles.
Methods:
Purchase tiles.

Transaction:

Responsibility: Represents financial transactions.
Association: Linked to TileOwnershipManager.

Property, TrainStation, UtilityTile, ChanceCardTile, DefaultTile, JailTile:

Responsibility: Concrete classes inheriting from Tile or implementing IOwnable.
Specifics: Each has attributes and methods relevant to their function in the game.

Design Choices

Encapsulation: The design encapsulates functionalities within their respective classes. For example, the MonopolyGame class encapsulates game control, and Player encapsulates player-specific actions.

Inheritance: Inheritance is used to extend the base Tile class into specific tile types. This allows for code reusability and a clear organizational structure, making it easy to add new tile types in the future.

Interfaces: The use of the IOwnable interface allows different types of tiles to be purchased and owned without needing to know their specific type. This demonstrates polymorphism and enhances flexibility in the design.

Aggregation: Aggregation is used to represent ownership, where a Player doesn't just associate with Tile objects but contains references to specific Tile objects. This enhances clarity in representing the 'has-a' relationship.