

РУСЕНСКИ УНИВЕРСИТЕТ „АНГЕЛ КЪНЧЕВ“

Катедра: Компютърни системи и технологии

Дисциплина: Информационни системи

КУРСОВА ЗАДАЧА

**По дисциплината
„Информационни системи“**

Изготвили:

Бисер Ценков

Борислав Борисов

Кристиян Петров

Мирослав Павлов

Приел:

/гл. ас. д-р Цв. Кънева/

фак. номера - 223029, 223028, 223015, 223053

спец. „Компютърни системи и технологии“

4-ти курс, 26А група

**Русе
2025**

1. Цели на проекта

Главната цел на нашият проект е да се проектира и разработи информационна система за управление на клуб по настолни игри. Игрите ще могат да се избират чрез използването на дигитални устройства на масите. Това ще улесни процесът на избор на подходяща настолна игра чрез филтриране по брой играчи, време за игра и ниво на трудност на самата игра. Друга важна цел е да се ограничи нуждата от постоянно съдействие от страна на служителите при препоръка и обяснение на основната информация за игрите. Един от стремежите на задачата е спокойствието както на служителя, така и на клиентите. Атмосферата трябва да е организирана по такъв начин, че в нея клиентите да могат самостоятелно и удобно да разглеждат наличните игри. Една от най-важните ни цели е да се осигури равен достъп до информация за хора с увреждания (хора с увреден слух, зрение, трудна подвижност т.н.) чрез текстово и визуално представяне на съдържанието. Автоматизация и управление на наличностите също е важен аспект за нас. Да се вижда актуална информация за заетите маси и за заетите игри в реално време. От гледна точка към клиентите проектът има задача да събира „feedback” с цел осъвършенстването и. Чрез него закупуването на нови игри за клуба ще е в пъти по-лесно, увеличавайки посещаемостта и клиентското удовлетворение. Уголемената разновидност на игри би помогнала за организацията на събития с новопривлечените клиенти.

2. Функционални изисквания

2.1 Управление на игри

- Системата трябва да позволява добавяне, редактиране, преглеждане и премахване настолни игри;
- Всяка игра трябва да има описание (брой играчи, време, за игра, сложност, жанр);
- Системата трябва да показва наличността на всяка игра (Свободна / Заета).

2.2 Избора на игра от клиент

- Клиентът трябва да може да разглежда игри чрез таблет на масата;
- Системата трябва да позволява филтриране на игри по брой играчи, време за игра, сложност, тип игра;
- Клиентът трябва да може да маркира игра като „избрана”.

2.3 Поддръжка на клиента

- Системата трябва да предоставя основна информация и правила за играта;
- Системата трябва да поддържа текстовото представяне на информацията ;
- Интерфейсът трябва да е достъпен без регистрация.

2.4 Административни функции

- Персоналът трябва да може да следи кои игри са заети;
- Системата трябва да позволява обновяване на статуса на игрите в реално време;
- Администраторът трябва да има достъп до статистика за използваемост на игрите.

3. Нефункционални изисквания

3.1 Използваемост

- Интерфейсът трябва да бъде интуитивен и лесен за използване;
- Системата трябва да бъде подходяща за потребители без технически умения или компютърна грамотност;
- Навигацията трябва да бъде лесна и логична.

3.2 Производителност

- Системата трябва да реагира на потребителски действия до 2 секунди или колкото се може по-бързо;
- Промените в наличността на игрите трябва да се отразяват в реално време;

3.3 Надеждност

- Системата трябва да работи стабилно по време на работното време на клуба;
- При прекъсване на връзката системата не трябва да губи данни (да има backup).

3.4 Сигурност

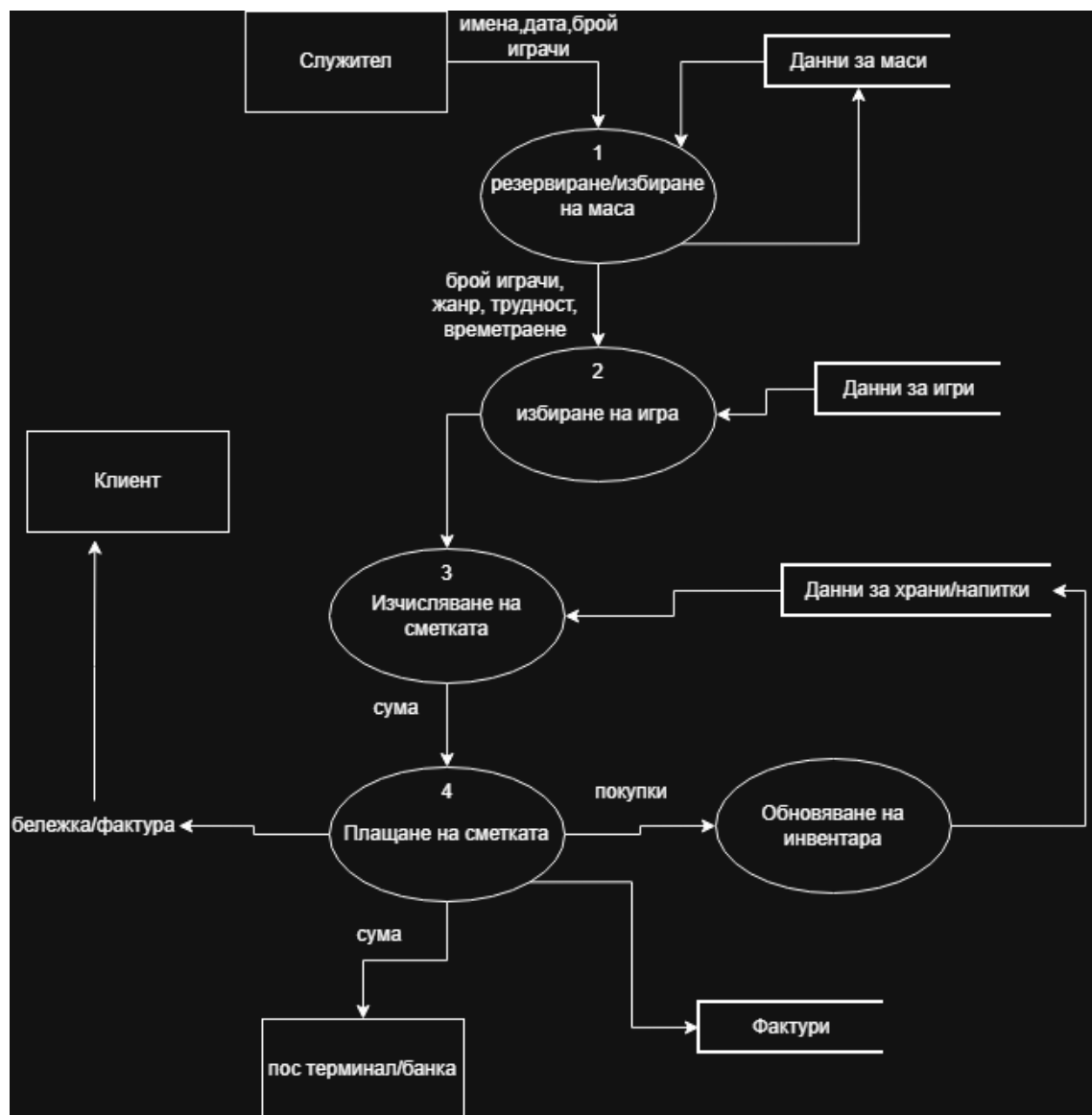
- Системата не трябва да изисква лични данни от клиентите с изключение feedback;
- Достъпът до административните функции трябва да бъде ограничен.

3.5 Достъпност

- Информацията трябва да бъде представена в четим текстов формат;
- Интерфейсът трябва да бъде подходящ за хора с увреждания;
- Системата трябва да поддържа поне един допълнителен език (примерно английски).

4. Моделиране на процесите и данните

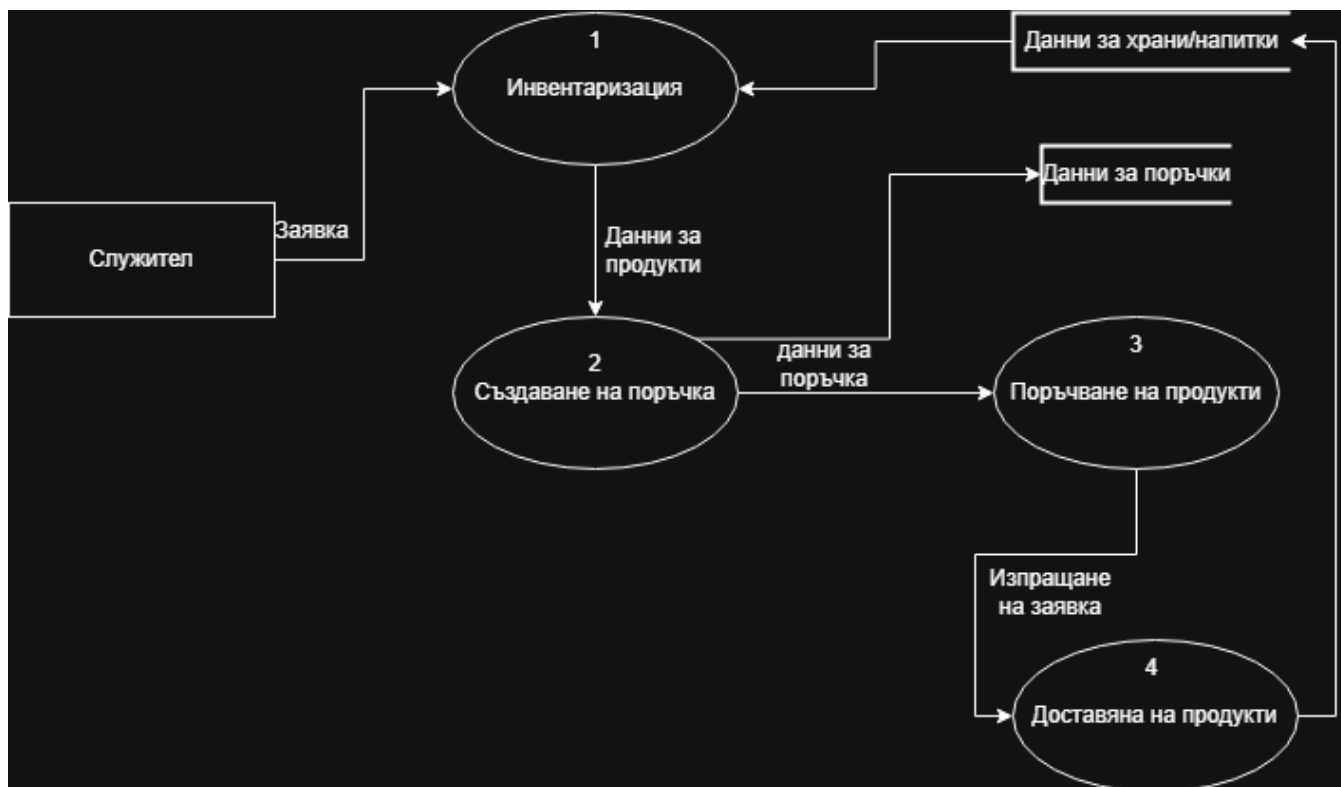
ДПД Диаграма 1



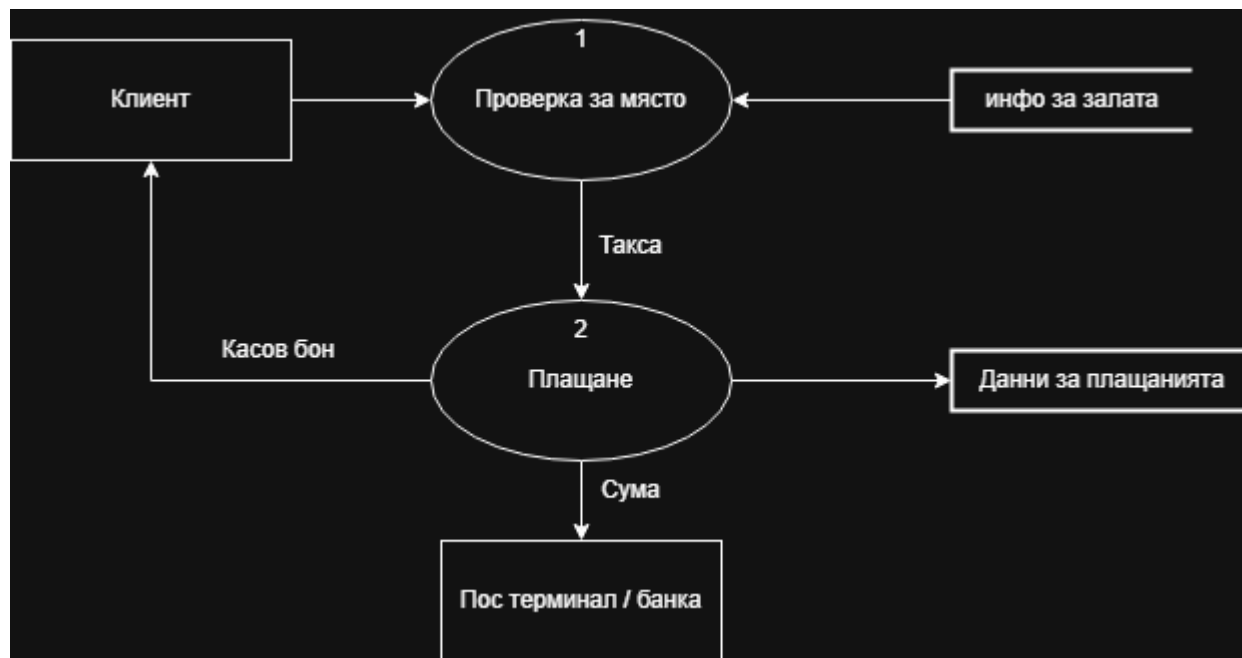
ДПД Диаграма 2



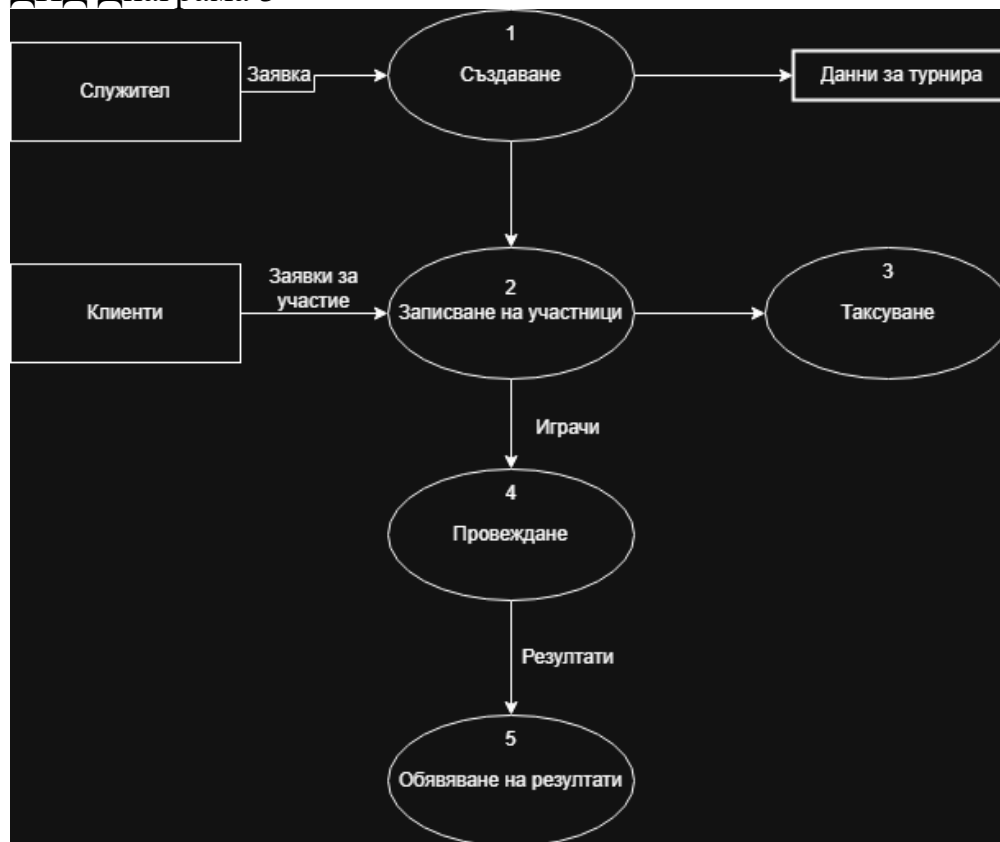
ДПД Диаграма 3



ДПД Диаграма 4



ДПД Диаграма 5



Релационна диаграма

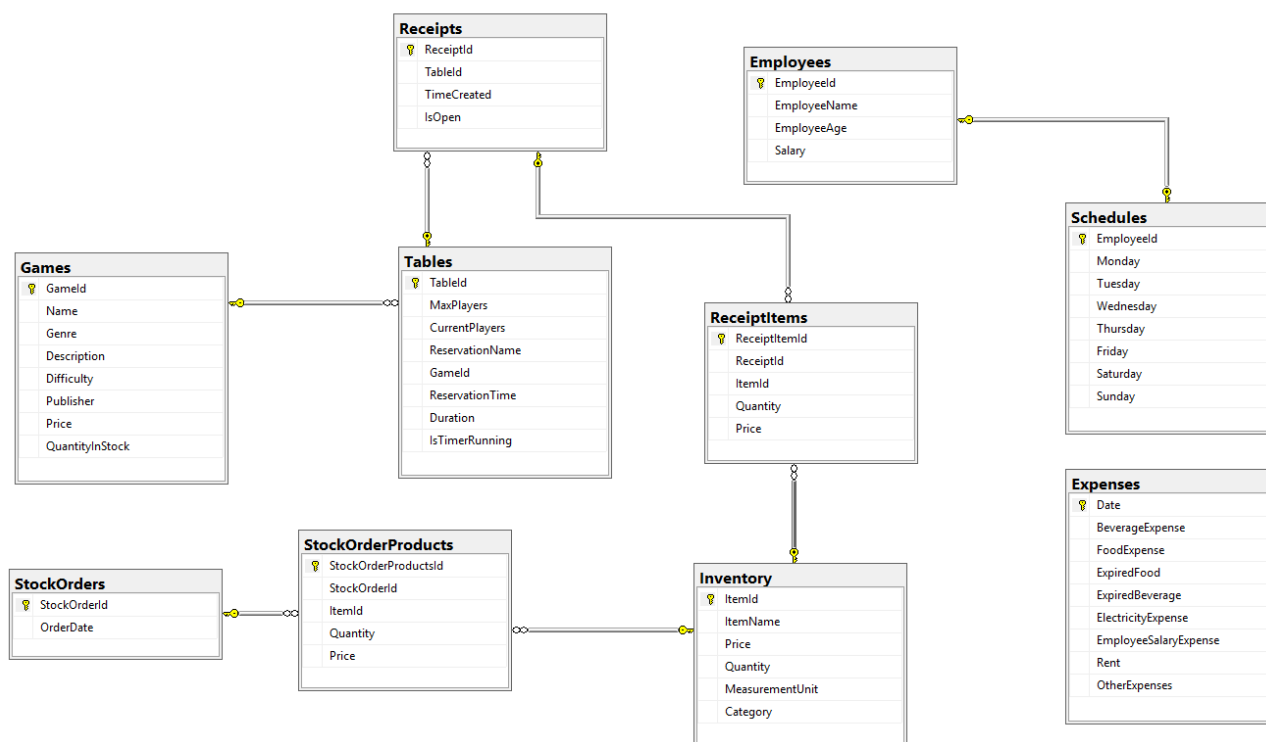
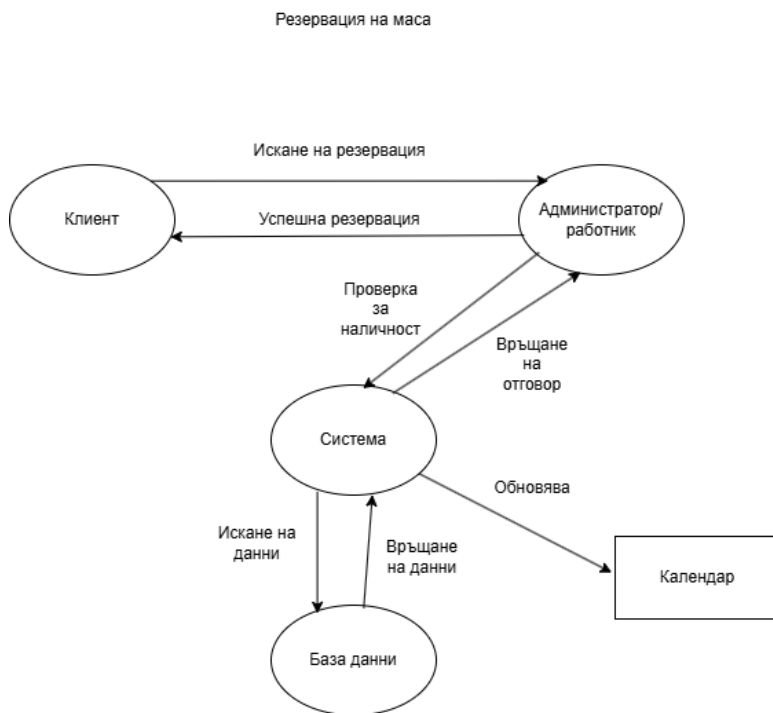


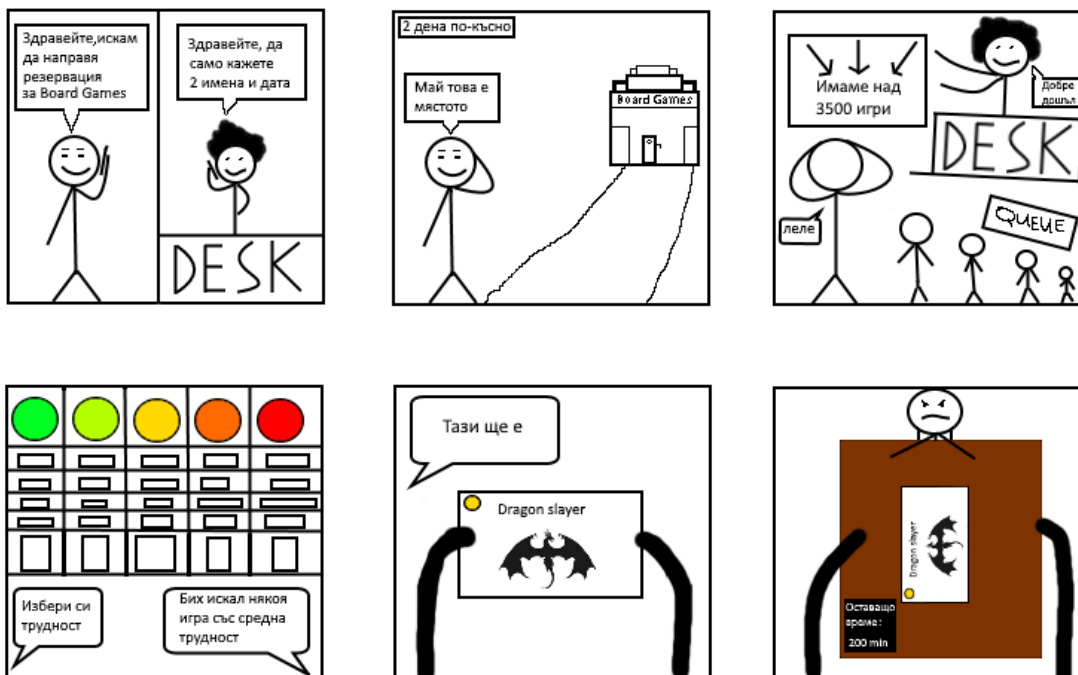
Таблица за вземане на решения

Условия/Действия	Правило 1	Правило 2	Правило 3	Правило 4	Правило 5	Правило 6	Правило 7	Правила	
Условия								Ако продуктът е наличен (Y), плащането е потвърдено (Y), тогава потвърди поръчката и изпрати продукта (X).	1
Играта е в наличност	Y	Y	N		Y	N		Ако играта е налична (Y), но плащането не е потвърдено (N), тогава изпрати уведомление за очакване на плащане (X).	2
Свободна маса				Y	N	N	Y	Ако играта не е налична (N), тогава предложи друга игра.(X)	3
Плащането е потвърдено	Y	N						Ако има свободна маса(Y) и е свободна избраната игра (Y), тогава се настаняват клиентите(X).	4
Играта е свободна				Y			N	Ако няма свободна маса(N) и е налична избраната игра (Y), тогава се предлага покупка(X) или отлагане за друг ден/час (резервация)(X).	5
Действия								Ако няма свободна маса(N) и не е налична/свободна избраната игра (N), тогава се отлага за друг ден/час (резервация) (X).	6
Потвърди поръчка и изпрати	X							Ако има свободна маса(Y), но не е свободна избрана игра (N), тогава се предлага друга игра(X).	7
Изчакване на заплащане		X							
Предложи друга игра			X				X		
Отложи за друг момент					X	X			
Настаняване на клиентите				X					
Предложение за покупка					X				

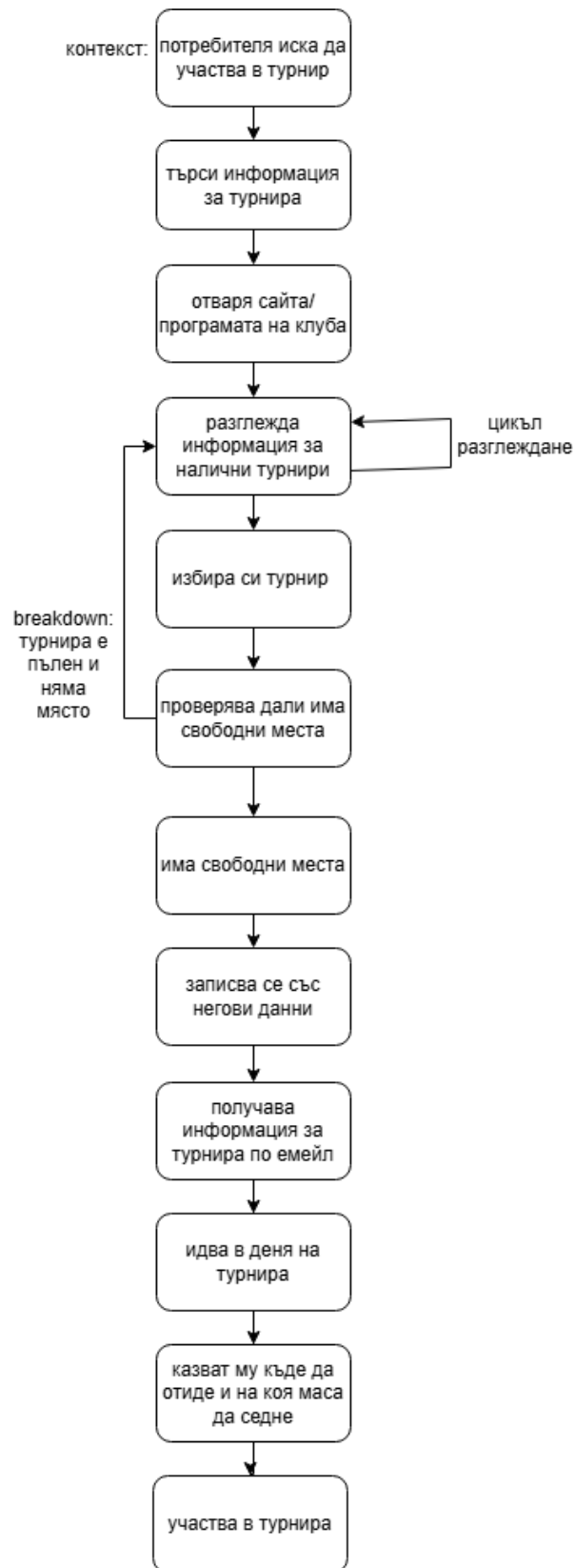
Flow model



Story board



Sequence model



5. Описание на алгоритмите - формулиране на проблема, псевдо –код

5.1 Управление на основното състояние

Проектът използва Entity Framework заявки за свързване към база данни. Системата проследява 6 маси чрез `_currentTableId` и поддържа една активна `_currentReceipt` за всяка маса, която се зарежда само за отворени бележки чрез заявка към базата данни с `Include` за свързаните `Items` и `Item` данни. Данните за масите (игри, продължителност, състояние на таймера) се зареждат чрез `LoadTable`, а глобален `DispatcherTimer` (с интервал 1 секунда) обновява UI-я за показване на времето, показващо колко време е играно от началото до текущият момент на избраната маса. Цветовете на бутоните отразяват състоянието на таймера (тъмнолилав за заета, розово за свободна) за всички маси.

5.2 Работен процес на маси и таймер

Избор на маса прави заявка към базата данни за масиве (`Tables`) с текуща игра (`CurrentGame`), зареждане на отворената бележка, показване на продължителността. Задаване на игра (по желание), стартиране на таймера, задаване `IsTimerRunning = true`, `ReservationTime = Now`; спиране се дефинира като добавяне на изминалото време към `Duration`, нулиране на флага. Таймерът следи колко време е изминало от началото до текущият момент, преизчислявайки `shownTime = Duration + (Now - ReservationTime`, ако таймерът работи) само за текущата маса. Плащането нулира състоянието на масата (`Duration = Zero`, таймерът изключен), след като се покажат разходите за артикули и време (7 лв/час).

5.3 Меню и логика на бележките

Бутоните за категории зареждат `Inventory` по „Food“ / „Drink“ или мапват игрите като псевдо-артикули. Добавяне на артикул проверка/създаване на отворена бележка; увеличаване на `Receipt Items.Quantity`, ако артикулът вече съществува, или добавяне на нов; намаляване на `Inventory.Quantity`; запис чрез `SaveChanges`.

Обновяването на UI-я проектира ReceiptItems към анонимен view model (име, количество, сума по ред), сумира стойностите за показване. Проверката на наличностите предотвратява продажба над наличното, но липсват транзакции за конкурентен достъп (потенциален race condition).

5.4 Псевдо Код

List<Game> games;

Правило 1-2:

function PurchaseGame()

 CheckGameInventoryAmount(gameName);

 if (CheckGameInventoryAmount(gameName) <= 0)

 return "Game is out";

 END IF

 if (MoneySent = false)

 return "Expect payment!";

 END IF

 return "Sent Game!";

END FUNCTION

function CheckGameInventoryAmount(gameName)

 for (int i = 0; i < games.Count(); i++)

 if (games[i].Name = gameName)

```
        return games[i].Amount;

    else

        return "Game doesn't exist";

    END IF

END FOR

END FUNCTION
```

Правило 3:

```
function ProposeGame()

    if (CheckGameInventoryAmount = 0)

        return "Propose something else"

    END IF

END FUNCTION
```

Правило 4-7:

```
function SeatClients(numberOfClients)

    if (TableAvailable(numberOfClients) = true AND GameChosen = true)

        return "Seat Customers";

    else if ()

        return "Offer purchase or reservation";

    end if

end function
```

```
else if ()  
  
    return "";  
  
else  
  
    return "";  
  
END IF  
  
END FUNCTION
```

6. Елементи на TPS/DSS/EIS

6.1 Използване на системата като TPS

Системата за настолните игри функционира като TPS, като обработва ежедневни операции в реално време. Тези операции са:

- Маркиране на настолните игри като свободни или заети е една операция;
- Регистриране на избора на игра от дадена маса;
- Проследяване на използването на игрите по време;
- Управление на резервации и освобождаване на игри.

Тези транзакции са рутинни, повтаряеми и изискват бърза и надеждна обработка.

6.2 Използване на системата като DSS

За да функционира като DSS тя подпомага вземането на решения чрез анализ на събраните данни. Системата може да предоставя, справки за най-често избирани игри, например. Също анализ на натовареността по дни и часове, информация за предпочитанията на клиентите (брой играчи, сложност, продължителност), данните за игри, които рядко се използват. Тази информация подпомага персонала при

решения, свързани с организацията, подбора на нови игри и оптимизацията на услугата.

6.3 Използване на системата като EIS

Като EIS системата предоставя обобщена и визуализирана информация за управлението. Чрез табло с ключови показатели могат да се следят – най-популярните игри за даден период, средна заетост на масите, тенденции в поведението на клиентите. Тази информация е представена в опростен и разбираем вид, което позволява бърза ориентация и стратегическо планиране от страна на мениджмънта.

7. Интервю

Мирослав Павлов и Силенджан Кадоолу

1. Какво представлява типичен работен ден за теб?

Отговор:

Да работя с клиенти, които се интересуват от настолни игри, да виждам усмивки по лицата на хора, да предавам атмосфера. Да чистя, правя сандвичи и коктейли също.

2. Кои са основните ти отговорности - обслужване на клиенти, препоръки за игри или поддръжка на пространството?

Отговор:

По принцип е да правя всичко, от обслужване до приготвяне на списъци за пазаруване. Всичко правя с изключение на маркетинга.

3. Колко често обясняваш правила на игри и как сесправяш с различни нива на опит на клиентите?

Отговор:

Тук вече зависи от компанията, трябва само с 2-3 изречения да разбереш хората в какво настроение са, да знаеш какво би им харесало и от визията и говора общо взето трябва да се знае, че тази компания би се изкефила повече на соур отколкото на катан примерно.

4. Би ли предпочел да има някакъв начин хората да могат лесно да прочетат коя игра е за начинаещи или по-напреднали без да трябва ти да обясняваш?

Отговор:

По скоро тук ако са нови хора ще им обясня, защото никой не се кефи да чете правилата на някоя игра. А и аз като им обяснявам се запознавам с тях, говорим си и най-важното ще се върнат отново, защото атмосферата, която понякога получават им харесва. Ако искат вече някоя сложна игра с над 40 мин обяснение...да се оправят както могат.

5. Ако дойде човек със затруднение (примерно нем или глух) как би се опитал да им помогнеш?

Отговор:

Ще се държа както с всеки един клиент. А на това как бих се опитал да помогна, ще се намери начин стига да попадна в тази ситуация.

6. Какво те привлече да работиш на тази локация?

Отговор: yugioh

7. Да кажем, че си човек, който до сега не е играл настолни игри. Колко време би ти отнело да се обучиш, като кажем, че има огромна наличност на разновидни игри, които ти трябва да можеш даобясниш на новодошли клиенти?

Отговор:

зависи от човека, играта и колко време му трябва да възприеме дадена игра. Всеки човек е различен и това е трудно да се каже, но лично за мен ми отне към 2 седмици

да науча 10 игри без да си давам зор. Буквално игрите не са сложни, сложното в тях е как мислиш над играта и колко време се нарежда.