

Popug Event Storming

Таск-трекер

1. Таск-трекер должен быть отдельным дашбордом и доступен всем сотрудникам компании UberPopug Inc.

Actor - Popug

Command - GetTasks

Data - Popug.Id

Event - Tasks.ListSent

2. Авторизация в таск-трекере должна выполняться через общий сервис авторизации UberPopug Inc (у нас там инновационная система авторизации на основе формы клюва).

Actor - Popug

Command - Auth

Data - Popug.BeakLength

Event - Popug.Authorized

3. В таск-трекере должны быть только задачи. Проектов, скоупов и спринтов нет, потому что они не умещаются в голове попуга.
4. Новые таски может создавать кто угодно (администратор, начальник, разработчик, менеджер и любая другая роль). У задачи должны быть описание, статус (выполнена или нет) и попуг, на которого заассайнена задача.

Actor - Popug

Command - CreateTask

Data - Task.Description, Task.Status, Task.Popug

Event - Task.Created, Task.Assigned

5. Менеджеры или администраторы должны иметь кнопку «заассайнить задачи», которая возьмёт все открытые задачи и рандомно заассайнит каждую на любого из сотрудников (кроме менеджера и администратора) . Не успел закрыть задачу до реассайна — сорян, делай следующую.

Actor - Popug

Command - ShuffleTask

Data - Popug.Role

Event - Task.Assigned

- a) Ассайнить задачу можно на кого угодно (кроме менеджера и администратора), это может быть любой существующий аккаунт из системы.
- b) Ассайнить задачу можно только кнопкой «заассайнить задачи»
- c) При нажатии кнопки «заассайнить задачи» все текущие не закрытые задачи должны быть случайным образом перетасованы между каждым аккаунтом в системе
- d) Мы не заморачиваемся на ограничение по нажатию на кнопку «заассайнить задачи». Её можно нажимать хоть каждую секунду.
- e) На одного сотрудника может выпасть любое количество новых задач, может выпасть ноль, а может и 10.
- f) Создать задачу не заассайненую на пользователя нельзя. Т.е. любая задача должна иметь попуга, который ее делает

- 6. Каждый сотрудник должен иметь возможность видеть в отдельном месте список заассайненных на него задач

Actor - Popug

Command - GetTasks

Data - Popug.Id

Event - Task.ListSent

- 7. Каждый сотрудник должен иметь возможность отметить задачу выполненной.

Actor - Popug

Command - CloseTask

Data - Task.Id, Task.Status

Event - Task.StatusChanged

Аккаунтинг: кто сколько денег заработал

- 1. Аккаунтинг должен быть в отдельном дашборде и доступным только для администраторов и бухгалтеров.

- a) у обычных попугов доступ к аккаунтингу тоже должен быть. Но только к информации о собственных счетах (аудит лог + текущий баланс).

Actor - Popug

Command - ShowAccount, ShowAuditLog

Data - Popug.Id, Popug.Role

Event - Account.Sent, AuditLog.Sent

2. У админов и бухгалтеров должен быть доступ к общей статистике по деньгами заработанным (количество заработанных топ-менеджментом за сегодня денег + статистика по дням).

Actor - Popug

Command - ShowStatistics

Data - Popug.Id, Popug.Role

Event - Statistics.Created

3. Авторизация в дашборде аккаунтинга должна выполняться через общий сервис аутентификации UberPopug Inc.

Actor - Popug

Command - Auth

Data - Popug.BeakLength

Event - Popug.Authorized

4. У каждого из сотрудников должен быть свой счёт, который показывает, сколько за сегодня он получил денег.

Actor - Popug

Command - ShowAccount

Data - Popug.Id, DateTime.Date

Event - Account.InfoSent

5. У счёта должен быть аудитлог того, за что были списаны или начислены деньги, с подробным описанием каждой из задач.

Actor - Account.Recalculated

Command - CreateAuditRecord

Data - Task.Id, Task.Description, DateTime.Now, Sum

Event - AuditLog.Created

4. Расценки:

- цены на задачу определяется единовременно, в момент появления в системе (можно с минимальной задержкой)
 - цены рассчитываются без привязки к сотруднику
 - формула, которая говорит сколько списать денег с сотрудника при ассайне задачи — `rand(-10..-20)$`
 - формула, которая говорит сколько начислить денег сотруднику для выполненной задачи — `rand(20..40)$`

Actor - Task.Created

Command - GenerateWriteOffPrice, GenerateAssessmentPrice

Data - Task.Id

Event - Price.Created

5. деньги списываются сразу после ассайна на сотрудника, а начисляются после выполнения задачи.

Actor - Task.StatusChanged, Task.Assigned

Command - AccountSumRecalculate

Data - Popug.Id, Task.Price

Event - Account.Rec Calculated

6. Дешборд должен выводить количество заработанных топ-менеджментом за сегодня денег.

а) т.е. сумма всех закрытых и заасайненных задач за день с противоположным знаком: `(sum(completed task amount) + sum(assigned task fee)) * -1`

Actor - Popug

Command - ShowFirmMoney

Data - Popug.Id, Popug.Role

Event - ???

7. В конце дня необходимо:

a) считать сколько денег сотрудник получил за рабочий день

Actor - EndOfDay event

Command - CalculateDayPayment

Data - Popug.Id

Event - Account.DayPaymentCalculated

b) отправлять на почту сумму выплаты.

Actor - Account.DayPaymentCalculated

Command - SendPaymentCalculatedMail

Data - Popug.Id, Popug.Email

Event - Email.SendMail

8. После выплаты баланса (в конце дня) он должен обнуляться,

Actor - Account.DayPaymentCalculated

Command - AccountReset

Data - Popug.Id

Event - Account.Reset

9. и в аудитлоге всех операций аккаунтинга должно быть отображено, что была выплачена сумма.

Actor - Account.Reset

Command - CreateAuditRecord

Data - Popug.Id, Account.DayPayment

Event - AuditLog.Created

10. отрицательный баланс переносится на следующий день. Единственный способ его погасить - закрыть достаточное количество задач в течение дня.

Actor - EndOfDay event

Command - CalculateDayPayment

Data - Popug.Id

Event - Account.NegativeBalanceDetected

11. Дашборд должен выводить информацию по дням, а не за весь период сразу.

а) вообще хватит только за сегодня (всё равно попуги дальше не помнят), но если чувствуете, что успеете сделать аналитику за каждый день недели — будет круто

Actor - Popug

Command - ShowAccount

Data - Popug.Id, Popug.Role

Event - Account.Sent

Аналитика

1. Аналитика — это отдельный дашборд, доступный только админам.

Actor - Popug

Command - Auth

Data - Popug.BeakLength

Event - Popug.Authorized

2. Нужно указывать, сколько заработал топ-менеджмент за сегодня и сколько попугов ушло в минус.

Actor - EndOfDay

Command - CreateStatistics

Data - DateTime.Today

Event - Statistics.Created

Actor - Popug

Command - ShowStatistics

Data - Popug.Id, Popug.Role

Event - Statistics.Sent

3. Нужно показывать самую дорогую задачу за день, неделю или месяц.

а) самой дорогой задачей является задача с наивысшей ценой из списка всех закрытых задач за определенный период времени

б) пример того, как это может выглядеть:

03.03 — самая дорогая задача — 28\$

02.03 — самая дорогая задача — 38\$

01.03 — самая дорогая задача — 23\$

01-03 марта — самая дорогая задача — 38\$

Actor - EndOfDay

Command - FindMostExpensiveTask

Data - TimePeriod

Event - MostExpensiveTask.Created

Actor - Popug

Command - ShowMostExpensiveTask

Data - TimePeriod

Event - MostExpensiveTask.Sent