

```

ui <- fluidPage(

add_busy_spinner(spin = "fading-circle", height = "15px"),

  navbarPage("Stochastický model revenue managementu - ver. 1.14, (c) Martin Petricek",

    tabPanel("Úvodní informace",
      mainPanel(

        tags$div(

          tags$p("Vítejte v aplikaci prezentující model revenue managementu určený pro podniky či jednotlivce podnikající v segmentu ubytovacích služeb.")

        ),

        tabsetPanel(
          tabPanel("Jak aplikaci využívat",

            tags$div(

              tags$p("Při práci s aplikací je nutné vždy plně respektovat zadání a řídit se návodnými texty zejména při importování vlastních dat. Bez dodržování těchto zásad může celá aplikace přestat fungovat a bude vyžadovat restartování. Aplikace je rozdělena do čtyř základních etap, označeny jako: ZA - základní analýza, SM - stanovení množství, SC - stanovení ceny, PO - postoptimalizace. Při práci s aplikací je vhodné postupovat podle jednotlivých záložek tak, jak jsou řazeny."),

              tags$p("1. ZA - základní analýza. V této části se zadávají základní údaje, které následně slouží pro další zpracování celého modelu. Základní předpoklad je znalost jednotlivých zákaznických segmentů, neboť právě pro jednotlivé segmenty se také často výpočty realizují. V rámci základní analýzy (ZA) je stanovena cenová elasticita poptávky (buď metodou regresní analýzy nebo simulací s využitím obloukové metody), predikováno poptávané množství (aplikace sama zvolí nejvhodnější parametry modelu predikce na základě historických dat) na další období, stanovena základní hodnota variabilních nákladů a fixních nákladů na základě regresní analýzy."),

              tags$p("2. SM - stanovení množství. Tato část v sobě implementuje zejména určení tzv. protection limits a booking limits pro jednotlivé segmenty. Jedná se tedy o doporučení, jak velké kapacity vyčlenit pro jednotlivé segmenty s předpokladem, že některý segment je ochotný za danou kapacitu akceptovat vyšší cenu. Kromě protection a booking limits je možné využít také doporučení určení hodnoty overbookingu na základě stochastické optimalizace."),

              tags$p("3. SC - stanovení ceny. Část stanovení ceny je rozdělena na dvě položky. Nejprve je nutné stanovit minimální prodejní cenu a poté je možné soustředit se na

```

optimalizaci. Optimalizace ceny je dvoufázová a probíhá na základě: 1. určení základní nabídkové ceny (jedná se o doporučení na základní cenu, která vychází z optimalizačního procesu postaveném na stochastické optimalizaci s využitím znalosti očekávané cenové elasticity poptávky daného segmentu) a 2. stanovení ceny konkrétní rezervace (ta v sobě odráží komplexní set prvků, které cenu ovlivňují)."),

tags\$p("4. PO - postoptimalizace. Záložka soustředí se na postoptimalizaci v sobě jednak schraňuje veškeré údaje za všechny segmenty a je možné ji tedy využít jako záložku, kam se zapisují jednotlivé údaje pro dílčí segmenty. Také však slouží jako podklad pro citlivostní analýzu a postoptimalizační simulaci overbookingu. Citlivostní analýza je klíčovým nástrojem pro hlubší analytické úvahy."),

tags\$p("\_\_\_\_\_"),

tags\$p("Některé optimalizační výpočty jsou výpočetně velice náročné a proto může aplikace tyto údaje zpracovávat delší dobu. Pokud je v pravém horním rohu aplikace zobrazena načítací animace, tak jsou prováděny výpočty."),

tags\$p("\_\_\_\_\_"),

tags\$p("Detailní popis jednotlivých kroků je uveden v autorově monografii."),  
tags\$p("V případě práce s aplikací v jiném prostředí je nutné mít nainstalované následující knihovny skriptovací jazyka R: shiny, ffp2, readxl, lubridate, nloptr, shinybusy a tidyverse.")

)  
)  
,

tabPanel("Licenční podmínky",  
tags\$div(  
tags\$p("Aplikace vznikla v prostředí skriptovacího jazyka R a tedy pod licenci  
AGPL."),

tags\$p("Respektujte prosím šíření kódu v rámci CC BY-NC-ND 4.0.")

)  
)  
)

tabPanel("O autorovi",  
tags\$div(  
tags\$p("Autor: Ing. Martin Petříček, Ph.D."),  
tags\$p("e-mail: m.petricek@email.cz"),  
tags\$p(tags\$a(href="http://www.martinpetricek.eu", "webové stránky autora")),  
tags\$p("Aplikace vznikla jako výsledek několikaletého výzkumu v oblasti  
Revenue Managementu v segmentu ubytovacích služeb. Jedná se o výsledek práce celého  
týmu soustředícího se kolem autorovi osoby. Díky patří zejména Ing. Štěpánovi Chalupovi  
za dlouhodobou a ukázkovou spolupráci, Ing. Jiřině Jenčkové za praktické připomínky a  
komentáře, Ing. Radce Telyčkové za praktické postřehy, Ilkim Schuster, B.A. za sdílení

zkušeností, Andreasu Schusterovi za podporu při vývoji a dalším kolegyním a kolegům, kteří se přímo či nepřímo účastnili několika diskuzí, kulatých stolů, vyplňování dotazníků atp. Všichni se tak podílejí na rozvoji Revenue Managementu v oblasti ubytovacích služeb v České republice a na Slovensku." ),

tags\$p("Veliké díky patří mojí milované manželce za obrovskou podporu a hluboké pochopení při prozkoumávání slepých uliček při vývoji této aplikace.")

)

)

)

)

),

tabPanel("ZA - Cenová elasticita poptávky - regrese",

tags\$div(

tags\$p("Hodnota cenové elasticity poptávky (Epd) je v tomto panelu stanovena pomocí log-log regresní analýzy. Proto je nutné respektovat správné vložení dat dle pokynů níže. Před interpretací samotných hodnot je nutné ověření kvality regresního modelu.")

),

mainPanel(

tabsetPanel(

tabPanel("Vstupní hodnoty"

fileInput("souborR", "Nahrát soubor ve formátu \*.xls, ve kterém jsou zaplněny první dva sloupce hodnotami QD a ADR, tedy poptávaným množstvím a tomu odpovídající hodnotě ADR."),

dataTableOutput("vstupyR")

),

tabPanel("Regrese",

actionButton("regrese", "Proveď regresi"),

verbatimTextOutput("resultR"),

plotOutput("regreseplot")

),

tabPanel("Koeficient cenové elasticity",  
verbatimTextOutput("resepdR"),

tags\$div(

tags\$p("Hodnoty elasticity a jejich interpretace:"),

tags\$p("- záporné hodnoty elasticity značí tradiční průběh poptávkové křivky (s růstem ceny klesá poptávané množství"),

tags\$p("- koeficient elasticity v absolutní hodnotě vyšší než jedna značí cenově elastický segment"),

tags\$p("- koeficient elasticity v absolutní hodnotě nižší než jedna značí cenově neelastický segment"),

```

tags$p("- kladné hodnoty elasticity značí situaci tzv. Giffenova paradoxu")
)
)
)
)
),

```

#první velký panel nahoře obsahující sidebarpanel a další tabpanely v tabsetpanelu (ty jsou všechny v mainpanelu):

```

tabPanel("ZA - Cenová elasticita poptávky - oblouková",
tags$div(
tags$p("V tomto panelu je stanovena hodnota koeficientu cenové elasticity poptávky
(Epd) s využitím obloukové metody v kombinaci s Monte Carlo simulací. Výsledná hodnota
je tak lépe znázorněna v histogramu a je vhodné respektovat celé rozložení, než pouze
jednu výstupní hodnotu v podobě např. střední hodnoty.")
),

```

```

sidebarPanel(
numericInput("P1", "Původní cena:", 0),
numericInput("P2", "Nová cena", 0),
numericInput("Q1", "Původní prodávané množství:", 0),
numericInput("Q2", "Očekávané prodávané množství:", 0),
actionButton("run", "Vypočti cenovou elasticitu")
),

```

```

mainPanel(
tabsetPanel(
tabPanel("Výsledky",
(h3("Odhadnutá hodnota elasticity")),
textOutput("Epd")
),
tabPanel("Graf",
(h3("Histogram")),
plotOutput("Epdplot")
)
)
),
),

```

#třetí velký panel nahoře:

```

tabPanel("ZA - Predikce poptávaného množství",
tags$div(

```

tags\$p("Predikované poptávané množství (Qd) je generováno pomocí automatického výběru nejvhodnějších parametrů modelu SARIMA. Vstupní data musí obsahovat pouze údaje o poptávaném množství na denní bázi. Výpočetně náročný výstup je poté generován automaticky.")

```
),  
  
mainPanel(  
  tabsetPanel(  
    tabPanel("Vstupní hodnoty",  
      fileInput("soubor", "Nahrát soubor ve formátu *.xls který obsahuje zaplněný  
jeden sloupec (označený jako Qd) a to hodnotami poptávaného množství na denní bázi."),  
      dataTableOutput("vstupy")  
    ),  
    tabPanel("Predikované hodnoty",  
      dateInput("startdatum", "Datum začátku predikce (den v týdnu musí být  
následující po poslední hodnotě vstupních dat)"),  
      actionButton("predikuj", "Proveď predikci"),  
      dataTableOutput("fcresult")  
    ),  
    tabPanel("Graf",  
      plotOutput("fcplot")  
    )  
  )  
)  
),
```

# čtvrtý velký panel nahoře:

```
tabPanel("ZA - Predikce - analýza výsledků",  
  tags$div(  
    tags$p("Panel slouží jako analýza predikce z předcházejícího výstupu. Je nutné mít  
nejprve provedenou predikci v předchozím kroku a poté je možné vygenerovat výstupy v  
tomto panelu.")  
  ),  
  mainPanel(  
    tabsetPanel(  
      tabPanel("Výstup z predikce",  
        actionButton("ukazttabulku", "Zobraz"),  
        tableOutput("ttabulka")  
      ),  
      tabPanel("Data z predikce",  
        verbatimTextOutput("datapredikce")  
      ),  
      tabPanel("Střední hodnota, směrodatná odchylka",  
        tags$div(  
          tags$p("Střední hodnota")
```

```

    ),
    verbatimTextOutput("strhod"),

    tags$div(
      tags$p("Směrodatná odchylka")
    ),
    verbatimTextOutput("smodch")
  )
)
)
),

```

# devátý panel - stanovení FC

```

  tabPanel("ZA - Stanovení FC",
    tags$div(

      tags$p("Hodnotu fixních nákladů je vhodné zadávat jako denní údaj. Je nutné
      respektovat formát vložení, který je předepsaný u vstupních hodnot. ")
    ),

    mainPanel(
      tabsetPanel(
        tabPanel("Vstupní hodnoty",
          fileInput("souborFC", "Nahrát soubor ve formátu *.xls, který obsahuje dva
          sloupce. V prvním sloupci (značeném jako R) jsou hodnoty tržeb v daném období a ve
          druhém sloupci (označeném jako TC) jsou hodnoty celkových nákladů odpovídající daným
          tržbám."),
          dataTableOutput("vstupyFC")
        ),
        tabPanel("Odhad FC",
          actionButton("regreseFC", "Odhadni FC"),
          verbatimTextOutput("resultFC"),
          plotOutput("regreseFCplot")
        ),
        tabPanel("Hodnota FC",

          tags$div(
            tags$p("Odhad hodnoty fixních nákladů. V případě záporné hodnoty je
            nutné zvolit jiný regresní model. Výsledný údaje je vhodné zapsat do záložky PO - Celkové
            hodnoty a postoptimalizační proces.")
          ),
          verbatimTextOutput("resFC")
        )
      )
    )
  ),

```

# desátý panel - stanovení AVC

```
tabPanel("ZA - Stanovení AVC",  
tags$div(  
    tags$p("Hodnoty variabilních nákladů se zadávají jako denní údaj. Není nutné vyplňovat všechny hodnoty.")  
    ),  
    sidebarPanel(  
        numericInput("AVCnarez", "Náklady na rezervaci", 0),  
        numericInput("AVCuklid", "Náklady na úklid", 0),  
        numericInput("AVCFaB", "Náklady Food and Beverages oddělení", 0),  
        numericInput("AVCudrzba", "Náklady na údržbu", 0),  
        numericInput("AVCmat", "Materiálové náklady ostatní", 0),  
        numericInput("AVCmzda", "Mzdové variabilní náklady", 0),  
        numericInput("AVCenerg", "Očekávané náklady na energii", 0),  
        numericInput("AVCost", "Ostatní variabilní náklady", 0),  
        actionButton("AVCrun", "Hodnota AVC")  
    ),  
    mainPanel(  
        (h3("Průměrné variabilní náklady")),  
        textOutput("AVCoutput")  
    )  
    ),  
    ),
```

# pátý velký panel nahoře:

```
tabPanel("SM - Určení protection limits pro segment",  
tags$div(  
    tags$p("Samotné stanovení protection limits je stanoveno na základě Monte Carlo simulace a to vždy pro segment, pro který byla provedena predikce.")  
    ),  
    ),
```

```

mainPanel(
  tags$div(

    tags$p("Před stanovením protection limits je nutné provést predikci poptávaného množství pro daný segment. Aplikace využívá data z predikce pro stanovení protection a booking limits pro jednotlivé segmenty."),
    tags$p("Zadávají se údaje v měně, ve které je prováděna tvorba cen."),
    tags$p("Výsledné hodnoty Booking Limit je vhodné zapsat do záložky PO - Celkové hodnoty a postoptimalizační proces.")
  ),

  sidebarPanel(
    numericInput("ValH", "Vyšší hodnota (Ochota zaplatit vyššího segmentu):", 0),
    numericInput("ValL", "Nižší hodnota (Ochota zaplatit analyzovaného segmentu):", 0),
    actionButton("PL", "Hodnota PL")
  ),
  mainPanel(
    plotOutput("PLplot"),
    tags$div(
      tags$p("Doporučená hodnota Protection Limit pro daný segment. Vždy je nutné při rozhodnutí respektovat celé pravděpodobnostní rozdělení z výše provedené optimalizace.")
    ),
    verbatimTextOutput("PLhodnoty"),
    (h3("Hodnota BL (Booking Limit) se stanoví jako rozdíl hodnot: kapacita - Protection Limit"))
  )
),

# šestý velký panel (zde bude totéž co EMSR, ale z panelu Overbookingu):

tabPanel("SM - Overbooking pro segment",
  tags$div(
    tags$p("Overbooking představuje doporučenou hodnotu na základě vstupních údajů, která je stanovena v počtu ubytovacích kapacit, nikoli v procentech. Je možné provádět a zkoušet různá nastavení na základě vstupních hodnot tak, aby bylo dosaženo optimálního výsledku. Závěr je prezentován jako výstup simulace Monte Carlo, což umožňuje hlubší pohled na danou problematiku i na extrémné hodnoty, kterých je možné teoreticky dosáhnout. Nastavení je vhodné kombinovat s postoptimalizační analýzou overbookingu.")
  ),

  mainPanel(

```



```

tags$div(
  tags$p("Doporučená hodnota Overbookingu je stanovena jako celé
pravděpodobnostní rozdělení na základě optimalizace. Hodnota je uvedena v počtu
ubytovacích kapacit.")
),
  sidebarPanel(
    numericInput("capOB", "Booking Limit:", 0),
    numericInput("MROB", "Průměrná cena:", 0),
    numericInput("costOB", "Náklady na stěhování:", 0),
    numericInput("strhodOB", "Průměrný počet no-shows a storno rezervací:", 0),
    numericInput("smodchOB", "Možná odchylka od výše uvedené hodnoty (v
množství rezervací):", 0),
    actionButton("OB", "Vypočti hodnotu OB")
  ),
  mainPanel(
    plotOutput("OBplot"),
    verbatimTextOutput("OBhodnota")
  )
)
),

```

# jedenáctý panel - stanovení minimální ceny

```

tabPanel("SC - Minimální prodejní cena",
  tags$div(
    tags$p("Hodnota minimální prodejní ceny vychází z několika vstupních údajů a
kombinuje v sobě jak nastavené podnikové cíle (v podobě ukazatele ROS), tak i ostatní
parametry. Údaje o nákladech se zadávají na denní bázi.")
  ),

```

```

  sidebarPanel(
    numericInput("PminBL", "Booking limit", 0),
    numericInput("Pmincap", "Kapacita zařízení", 0),
    numericInput("PminFC", "Hodnota fixních nákladů", 0),
    numericInput("PminAVC", "Hodnota průměrných variabilních nákladů", 0),
    numericInput("Pminocc", "Průměrná obsazenost segmentu (desetinné číslo
vyjadřující procenta)", 0),
    numericInput("PminROS", "Cílová hodnota ROS (desetinné číslo vyjadřující
procenta)", 0),
    actionButton("Pminrun", "Minimální cena")
  ),
  mainPanel(

```

```

        (h3("Minimální prodejní cena")),
        textOutput("Pminres")
    )
),

```

# osmý panel

```

    tabPanel("SC - Optimalizace ceny",
        tags$div(
            tags$p("Panel, který se soustředí na optimalizaci ceny ji provádí jako dvoufázovou. Nejprve je určena tzv. základní nabídková cena. Tato cena v sobě kombinuje optimalizaci na základě znalosti cenové elasticity poptávky a dalších parametrů zadaných v předcházejících krocích spolu s Monte Carlo simulací. Jako výstupy tedy není jedna hodnota, ale celé rozdělení, se kterým je vhodné také patřičně pracovat zejména při určování cenových limitů. Cena konkrétní rezervace poté představuje druhou fázi cenové optimalizace a představuje veškeré vstupy, které je vhodné zakomponovat do tvorby ceny.")
        ),

    mainPanel(
        tabsetPanel(
            tabPanel("Základní nabídková cena",
                sidebarPanel(
                    numericInput("PLRV", "Referenční cena:", 0),
                    numericInput("PEPD", "Hodnota cenové elasticity:", 0),
                    numericInput("PQD", "Očekávané prodeje:", 0),
                    numericInput("Pcost", "AVC", 0),
                    actionButton("Prun", "Základní nabídková cena")
                ),
                mainPanel(
                    plotOutput("Pplot"),
                    verbatimTextOutput("Phodnoty"),
                    tags$div(
                        tags$p("Doporučená změna v % oproti referenční ceně")
                    ),
                    textOutput("Pdoporuceni")
                )
            ),
            tabPanel("Cena konkrétní rezervace",
                tags$div(
                    tags$p("Není nutné vyplňovat všechna pole.")
                ),

            sidebarPanel(
                numericInput("Pznc", "Základní nabídková cena", 0),
                numericInput("Pbmin", "Minimální cena", 0),

```



),

```
mainPanel(  
  tabsetPanel(  
    tabPanel("Vstupy",  
      navlistPanel(  
        tabPanel("segment 1",  
          numericInput("S1BL", "Booking Limit", 0),  
          numericInput("S1OB", "Overbooking", 0),  
          numericInput("S1Epd", "Cenová elasticita", 1),  
          numericInput("S1P", "Prodejní cena", 1),  
          numericInput("S1Q", "Predikované poptávané množství", 1)  
        ),  
        tabPanel("segment 2",  
          numericInput("S2BL", "Booking Limit", 0),  
          numericInput("S2OB", "Overbooking", 0),  
          numericInput("S2Epd", "Cenová elasticita", 1),  
          numericInput("S2P", "Prodejní cena", 1),  
          numericInput("S2Q", "Predikované poptávané množství", 1)  
        ),  
        tabPanel("segment 3",  
          numericInput("S3BL", "Booking Limit", 0),  
          numericInput("S3OB", "Overbooking", 0),  
          numericInput("S3Epd", "Cenová elasticita", 1),  
          numericInput("S3P", "Prodejní cena", 1),  
          numericInput("S3Q", "Predikované poptávané množství", 1)  
        ),  
        tabPanel("segment 4",  
          numericInput("S4BL", "Booking Limit", 0),  
          numericInput("S4OB", "Overbooking", 0),  
          numericInput("S4Epd", "Cenová elasticita", 1),  
          numericInput("S4P", "Prodejní cena", 1),  
          numericInput("S4Q", "Predikované poptávané množství", 1)  
        ),  
        tabPanel("segment 5",  
          numericInput("S5BL", "Booking Limit", 0),  
          numericInput("S5OB", "Overbooking", 0),  
          numericInput("S5Epd", "Cenová elasticita", 1),  
          numericInput("S5P", "Prodejní cena", 1),  
          numericInput("S5Q", "Predikované poptávané množství", 1)  
        ),  
        tabPanel("Celkové údaje",  
          numericInput("sumFC", "Fixní náklady", 0),  
          numericInput("sumAVC", "AVC", 0)  
        )  
      )  
    )  
  )  
)
```

(c) Martin Petricek (draft version of the book) **FORM (TO) Stochastic Revenue Management model**



```
server <- function (input, output, session) {
```

```
### 1. panel (elasticita)
```

```
#údaje pro action button
rvepd <- reactiveValues(data = NULL)
observeEvent(input$run,{
  output$Epd <- renderText({
    (((input$Q2 - input$Q1)) / (input$Q1 + input$Q2)) / ((input$P2 - input$P1) / (input$P1
+ input$P2)))
  })

  output$Epdplot <- renderPlot({
    opakuj <- replicate (n = 1000, {
      epd <- (((rnorm(n = 1000, input$Q2, 0.5)) - input$Q1) / (input$Q1 + (rnorm(n = 1,
input$Q2, 0.5)))) / ((input$P2 - input$P1) / (input$P1 + input$P2)))
    })
    elasticita <- as.numeric(opakuj)
    hist(elasticita, main = paste("Cenová elasticita poptavky (Epd)", breaks = 100, col =
"grey", freq = F, xlim = c(min(elasticita), max(elasticita)))

  })

})
```

```
### 2. panel (elasticita - regrese)
```

```
# vstupní hodnoty nahrané ze souboru (není pod action buttonem)
```

```
output$vsstupyr <- renderDataTable({
  if(is.null(input$souborR)){return()}
  input$souborR
  dotabulkyR <- input$souborR
  aaR <- read_xls(dotabulkyR$datapath)
})
```

```
#action button
```

```
rreg <- reactiveValues(data = NULL)
observeEvent(input$regrese,{

  #výpočty
  dataR <- read_xls(input$souborR$datapath)
  a <- log(dataR$ADR)
  b <- log(dataR$QD)
  lregrese <- lm (a ~ b)
  hodnotyR <- summary(lregrese)
  vysledkyR <- as.numeric(lregrese$coefficients)
  elasticitaR <- vysledkyR[c(2)]
```

```

#výstup data v tabulce a v grafu
output$resultR <- renderPrint({
  hodnotyR
})

output$regreseplot <- renderPlot({
  plot(a ~ b, xlab = "log Qd", ylab = "log Price")
})

#výstup hodnota koeficientu elasticity
output$resepdR <- renderPrint({
  elasticitaR
})

})

```

### ### 3. panel (predikce)

#vstupní hodnoty nahrané ze souboru (nespadá pod action button a rovnou se zobrazuje output)

```

output$vsstupy <- renderDataTable({
  if(is.null(input$soubor)){return()}
  input$soubor
  dotabulky <- input$soubor
  aa <- read_xls(dotabulky$datapath)
})

```

```

#action button #
rvpr <- reactiveValues(data = NULL)
observeEvent(input$predikuj,{

```

```

  #výpočty
  data <- read_xls(input$soubor$datapath)
  is.numeric(data)
  Qd <- ts(data, start = 1, frequency = 7)
  fsarima <- auto.arima(Qd, trace = TRUE, stepwise = FALSE, approximation = FALSE)
  fc <- forecast(fsarima, h=30, level = c(10, 15))
  fcreult <- round(fc$mean, d=0)

```

```

#výstup graf
output$fcplot <- renderPlot({
  autoplot(fc)
})

```

```

#výstup data v tabulce (predikované)
startdatum <- input$startdatum

```

```

den <- mday(startdatum + 0:29)
mesic <- months(startdatum + 0:29)
dny <- weekdays(startdatum + 0:29)
fcrezult_datumy <- cbind (den, mesic, dny, fcrezult) #tento řádek spojuje hodnoty
nadtím definované do jednoho - do jedné tabulky

```

```

output$fcrezult <- renderDataTable({
  fcrezult_datumy
})

```

#}) - toto je původní ukončení action button v 3. panelu (nakopírováno na konec panelu pět, aby se i tam zobrazovaly výsledky)

### 4. panel

```

#action button
rvtest <- reactiveValues(data = NULL)
observeEvent(input$ukazttabulku, {

  # výstup
  output$ttabulka <- renderTable({
    fcrezult_datumy
  })

  # druhý výstup
  output$datapredikce <- renderPrint({
    print(summary(fsarima))
  })

  # třetí výstup (střední hodnota)
  output$strhod <- renderPrint({
    fcstrhod <- mean(fcrezult)
    print(fcstrhod)
  })

  # čtvrtý výstup (směrodatná odchylka)
  output$smodch <- renderPrint({
    fcsmodch <- sd(fcrezult)
    print(fcsmodch)
  })

})

```

### 5. panel

```

#action button
rvpl <- reactiveValues(data = NULL)
observeEvent(input$PL, {

```



```

#výstup v mainpanelu graf (nejdřív výpočet poté zobrazit graf)

opakujPL <- replicate (n = 100, {
  x0 <- c(0)
  fn <- function(x) (-(1-rnorm(n = 5, mean(fcreport), sd(fcreport)+3))*input$ValH)
  hin <- function(x) (-(1-pnorm(x, mean(fcreport), sd(fcreport)+3))*input$ValH +
input$ValL)

  resPL <- auglag(x0, fn, gr = NULL, hin = hin)
  solution <- resPL$par
}, simplify = F)

resfinPL <- as.numeric(opakujPL)

output$PLplot <- renderPlot({
  hist(resfinPL, main = ("Hodnota Protection Limits"), breaks = 30, col = "grey", freq
= F)
})

#výstup v mainpanelu hodnoty

output$PLhodnoty <- renderPrint({
  mean(resfinPL)
})

}) #

### 6. panel

#action button
rvob <- reactiveValues(data = NULL)
observeEvent(input$OB, {

#výpočet OB
opakujOB <- replicate (n = 100, {
  x0 <- c(0)
  fn <- function(x) (input$costOB * rnorm(n = 2, input$strhodOB, input$smodchOB) * (x
- input$scapOB) + input$MROB * (1-rnorm(n = 2, input$strhodOB, input$smodchOB)) * (x -
input$scapOB))
  hin <- function(x) {

  h <- numeric(2)

```

```

    h[1] <- (input$costOB * rnorm(n = 1, input$strhodOB, input$smodchOB) * (x -
input$capOB) + input$MROB * (1-rnorm(n = 1, input$strhodOB, input$smodchOB)) * (x -
input$capOB))
    h[2] <- x
    return(h)
  }

  resOB <- auglag (x0, fn, gr = NULL, hin = hin)
  solution <- resOB$par
}, simplify = F)

no.rooms <- as.numeric(opakujOB)

#graf OB
output$OBplot <- renderPlot({
  hist(no.rooms, main = ("Overbooking"), breaks = 30, col = "grey", freq = F, xlim = c(0,
max(no.rooms)))
  curve(dnorm(x, mean = mean(no.rooms), sd = sd(no.rooms)), from = 0, to =
max(no.rooms), add = T, col = "red")
})

#hodnota OB

output$OBhodnota <- renderPrint({
  print(summary(no.rooms))
  #mean(no.rooms)
})
})

### 7. panel

#action button - analýza citlivosti
rvcit <- reactiveValues(data = NULL)
observeEvent(input$citlivost_run, {

  #výpočet analýzy citlivosti

  S1P <- input$S1P
  S1Q <- input$S1Q
  S2P <- input$S2P
  S2Q <- input$S2Q
  S3P <- input$S3P
  S3Q <- input$S3Q
  S4P <- input$S4P
  S4Q <- input$S4Q
  S5P <- input$S5P
  S5Q <- input$S5Q
  FC <- input$sumFC

```

```
AVC <- input$sumAVC
```

```
citl_analyza <- function(S1P = input$S1P,  
  S1Q = input$S1Q,  
  S2P = input$S2P,  
  S2Q = input$S2Q,  
  S3P = input$S3P,  
  S3Q = input$S3Q,  
  S4P = input$S4P,  
  S4Q = input$S4Q,  
  S5P = input$S5P,  
  S5Q = input$S5Q,  
  FC = input$sumFC,  
  AVC = input$sumAVC) {
```

```
  max_change <- 0.1
```

```
  changes <- seq(-max_change, max_change, by = 0.01)
```

```
  shock <- function(x) x * (1 + changes)
```

```
  #definování funkce zisku
```

```
  zisk <- function(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P, S4Q, S5P, S5Q, FC, AVC)  
    (S1P * S1Q + S2P * S2Q + S3P * S3Q + S4P * S4Q + S5P * S5Q) - FC - AVC *  
    (S1Q + S2Q + S3Q + S4Q + S5Q)
```

```
  #výpočet základního a updated zisku
```

```
  start_zisk <- zisk(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P, S4Q, S5P, S5Q, FC, AVC)
```

```
  zisk_new_S1P <- zisk(shock(S1P), S1Q, S2P, S2Q, S3P, S3Q, S4P, S4Q, S5P, S5Q,  
    FC, AVC)
```

```
  zisk_new_S1Q <- zisk(S1P, shock(S1Q), S2P, S2Q, S3P, S3Q, S4P, S4Q, S5P, S5Q,  
    FC, AVC)
```

```
  zisk_new_S2P <- zisk(S1P, S1Q, shock(S2P), S2Q, S3P, S3Q, S4P, S4Q, S5P, S5Q,  
    FC, AVC)
```

```
  zisk_new_S2Q <- zisk(S1P, S1Q, S2P, shock(S2Q), S3P, S3Q, S4P, S4Q, S5P, S5Q,  
    FC, AVC)
```

```
  zisk_new_S3P <- zisk(S1P, S1Q, S2P, S2Q, shock(S3P), S3Q, S4P, S4Q, S5P, S5Q,  
    FC, AVC)
```

```
  zisk_new_S3Q <- zisk(S1P, S1Q, S2P, S2Q, S3P, shock(S3Q), S4P, S4Q, S5P, S5Q,  
    FC, AVC)
```

```
  zisk_new_S4P <- zisk(S1P, S1Q, S2P, S2Q, S3P, S3Q, shock(S4P), S4Q, S5P, S5Q,  
    FC, AVC)
```

```
  zisk_new_S4Q <- zisk(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P, shock(S4Q), S5P, S5Q,  
    FC, AVC)
```

```
  zisk_new_S5P <- zisk(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P, S4Q, shock(S5P), S5Q,  
    FC, AVC)
```

```

zisk_new_S5Q <- zisk(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P, S4Q, S5P, shock(S5Q),
FC, AVC)
zisk_new_FC <- zisk(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P, S4Q, S5P, S5Q,
shock(FC), AVC)
zisk_new_AVC <- zisk(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P, S4Q, S5P, S5Q, FC,
shock(AVC))

```

```

diff_px <- function(px) px - start_zisk
diff_zisk_new_S1P <- diff_px(zisk_new_S1P)
diff_zisk_new_S1Q <- diff_px(zisk_new_S1Q)
diff_zisk_new_S2P <- diff_px(zisk_new_S2P)
diff_zisk_new_S2Q <- diff_px(zisk_new_S2Q)
diff_zisk_new_S3P <- diff_px(zisk_new_S3P)
diff_zisk_new_S3Q <- diff_px(zisk_new_S3Q)
diff_zisk_new_S4P <- diff_px(zisk_new_S4P)
diff_zisk_new_S4Q <- diff_px(zisk_new_S4Q)
diff_zisk_new_S5P <- diff_px(zisk_new_S5P)
diff_zisk_new_S5Q <- diff_px(zisk_new_S5Q)
diff_zisk_new_FC <- diff_px(zisk_new_FC)
diff_zisk_new_AVC <- diff_px(zisk_new_AVC)

```

#asi údaje pro graf

#výstup v podobě grafu

```
output$citlivost_plot <- renderPlot({
```

```
options(scipen = 999)
```

```
fmt <- function(x) format(x, big.mark = ",", scientific = FALSE)
```

```
msg <- paste("Výchozí hodnoty: S1P = ", fmt(S1P),
```

```
  ", S1Q = ", fmt(S1Q),
```

```
  ", S2P = ", fmt(S2P),
```

```
  ", S2Q = ", fmt(S2Q),
```

```
  ", S3P = ", fmt(S3P),
```

```
  ", S3Q = ", fmt(S3Q),
```

```
  ", S4P = ", fmt(S4P),
```

```
  ", S4Q = ", fmt(S4Q),
```

```
  ", S5P = ", fmt(S5P),
```

```
  ", S5Q = ", fmt(S5Q),
```

```
  ", FC = ", fmt(FC),
```

(c) Martin Petříček (draft version of the code) - STORM (Stochastic Revenue Management model)

```

", AVC = ", fmt(AVC))

plot(100 * changes, diff_zisk_new_S1P,
     xlab = "Změna vstupní proměnné (%)",
     ylab = "Změna zisku (CZK)",
     main = "Citlivostní analýza",
     type = "l",
     sub = msg)

lines(100 * changes, diff_zisk_new_S1Q, col = "red")
lines(100 * changes, diff_zisk_new_S2P, col = "blue")
lines(100 * changes, diff_zisk_new_S2Q, col = "green")
lines(100 * changes, diff_zisk_new_S3P, col = "azure")
lines(100 * changes, diff_zisk_new_S3Q, col = "cyan")
lines(100 * changes, diff_zisk_new_S4P, col = "coral")
lines(100 * changes, diff_zisk_new_S4Q, col = "gray")
lines(100 * changes, diff_zisk_new_S5P, col = "hotpink")
lines(100 * changes, diff_zisk_new_S5Q, col = "navy")
lines(100 * changes, diff_zisk_new_FC, col = "maroon")
lines(100 * changes, diff_zisk_new_AVC, col = "orchid")

#legenda grafu prej

leg_names <- c("S1P", "S1Q", "S2P", "S2Q", "S3P", "S3Q", "S4P", "S4Q", "S5P",
"S5Q", "FC", "AVC")
legend("bottomright",
      legend = leg_names,
      col = c("black", "red", "blue", "green", "azure", "cyan", "coral", "gray", "hotpink",
"navy", "maroon", "orchid"),
      pch = 15,
      bty = "n")

})

#
# citl_analyza()

citl_analyza()

})

# action button (cit analýza elasticity = citel)
rvcitel <- reactiveValues(data = NULL)
observeEvent(input$citel_run, {

#výpočty

```

```

S1P <- input$S1P
S1Q <- input$S1Q
S2P <- input$S2P
S2Q <- input$S2Q
S3P <- input$S3P
S3Q <- input$S3Q
S4P <- input$S4P
S4Q <- input$S4Q
S5P <- input$S5P
S5Q <- input$S5Q
S1Epd <- input$S1Epd
S2Epd <- input$S2Epd
S3Epd <- input$S3Epd
S4Epd <- input$S4Epd
S5Epd <- input$S5Epd

```

```

citl_analyza_citel <- function (S1P = input$S1P,
                               S1Q = input$S1Q,
                               S2P = input$S2P,
                               S2Q = input$S2Q,
                               S3P = input$S3P,
                               S3Q = input$S3Q,
                               S4P = input$S4P,
                               S4Q = input$S4Q,
                               S5P = input$S5P,
                               S5Q = input$S5Q,
                               S1Epd = input$S1Epd,
                               S2Epd = input$S2Epd,
                               S3Epd = input$S3Epd,
                               S4Epd = input$S4Epd,
                               S5Epd = input$S5Epd) {

```

```

  max_change_citel <- 0.1

```

```

  changes_citel <- seq(-max_change_citel, max_change_citel, by = 0.01)

```

```

  shock_citel <- function(x) x * (1 + changes_citel)

```

```

  #definování funkce tržeb s elasticitou

```

```

  trzby_citel <- function(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P, S4Q, S5P, S5Q, S1Epd,
                          S2Epd, S3Epd, S4Epd, S5Epd)

```

```

    ((S1Q + S1Epd * (1.05*S1P - S1P))*(S1Q / S1P)*S1P) + (S2Q + S2Epd * (1.05*S2P -
S2P)*(S2Q / S2P)*S2P) + (S3Q + S3Epd * (1.05*S3P - S3P)*(S3Q / S3P)*S3P) + (S4Q +
S4Epd * (1.05*S4P - S4P)*(S4Q / S4P)*S4P) + (S5Q + S5Epd * (1.05*S5P - S5P)*(S5Q /
S5P))*S5P)

```

#výpočet základních a update tržeb

```
start_trzby_citel <- trzby_citel(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P, S4Q, S5P, S5Q,  
S1Epd, S2Epd, S3Epd, S4Epd, S5Epd)
```

```
trzby_citel_new_S1P <- trzby_citel(shock_citel(S1P), S1Q, S2P, S2Q, S3P, S3Q, S4P,  
S4Q, S5P, S5Q, S1Epd, S2Epd, S3Epd, S4Epd, S5Epd)
```

```
trzby_citel_new_S1Q <- trzby_citel(S1P, shock_citel(S1Q), S2P, S2Q, S3P, S3Q, S4P,  
S4Q, S5P, S5Q, S1Epd, S2Epd, S3Epd, S4Epd, S5Epd)
```

```
trzby_citel_new_S2P <- trzby_citel(S1P, S1Q, shock_citel(S2P), S2Q, S3P, S3Q, S4P,  
S4Q, S5P, S5Q, S1Epd, S2Epd, S3Epd, S4Epd, S5Epd)
```

```
trzby_citel_new_S2Q <- trzby_citel(S1P, S1Q, S2P, shock_citel(S2Q), S3P, S3Q, S4P,  
S4Q, S5P, S5Q, S1Epd, S2Epd, S3Epd, S4Epd, S5Epd)
```

```
trzby_citel_new_S3P <- trzby_citel(S1P, S1Q, S2P, S2Q, shock_citel(S3P), S3Q, S4P,  
S4Q, S5P, S5Q, S1Epd, S2Epd, S3Epd, S4Epd, S5Epd)
```

```
trzby_citel_new_S3Q <- trzby_citel(S1P, S1Q, S2P, S2Q, S3P, shock_citel(S3Q), S4P,  
S4Q, S5P, S5Q, S1Epd, S2Epd, S3Epd, S4Epd, S5Epd)
```

```
trzby_citel_new_S4P <- trzby_citel(S1P, S1Q, S2P, S2Q, S3P, S3Q, shock_citel(S4P),  
S4Q, S5P, S5Q, S1Epd, S2Epd, S3Epd, S4Epd, S5Epd)
```

```
trzby_citel_new_S4Q <- trzby_citel(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P,  
shock_citel(S4Q), S5P, S5Q, S1Epd, S2Epd, S3Epd, S4Epd, S5Epd)
```

```
trzby_citel_new_S5P <- trzby_citel(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P, S4Q,  
shock_citel(S5P), S5Q, S1Epd, S2Epd, S3Epd, S4Epd, S5Epd)
```

```
trzby_citel_new_S5Q <- trzby_citel(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P, S4Q, S5P,  
shock_citel(S5Q), S1Epd, S2Epd, S3Epd, S4Epd, S5Epd)
```

```
trzby_citel_new_S1Epd <- trzby_citel(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P, S4Q,  
S5P, S5Q, shock_citel(S1Epd), S2Epd, S3Epd, S4Epd, S5Epd)
```

```
trzby_citel_new_S2Epd <- trzby_citel(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P, S4Q,  
S5P, S5Q, S1Epd, shock_citel(S2Epd), S3Epd, S4Epd, S5Epd)
```

```
trzby_citel_new_S3Epd <- trzby_citel(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P, S4Q,  
S5P, S5Q, S1Epd, S2Epd, shock_citel(S3Epd), S4Epd, S5Epd)
```

```
trzby_citel_new_S4Epd <- trzby_citel(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P, S4Q,  
S5P, S5Q, S1Epd, S2Epd, S3Epd, shock_citel(S4Epd), S5Epd)
```

```
trzby_citel_new_S5Epd <- trzby_citel(S1P, S1Q, S2P, S2Q, S3P, S3Q, S4P, S4Q,  
S5P, S5Q, S1Epd, S2Epd, S3Epd, S4Epd, shock_citel(S5Epd))
```

```
diff_px_citel <- function(pxcitel) pxcitel - start_trzby_citel
```

```
diff_trzby_citel_new_S1P <- diff_px_citel(trzby_citel_new_S1P)
```

```
diff_trzby_citel_new_S1Q <- diff_px_citel(trzby_citel_new_S1Q)
```

```
diff_trzby_citel_new_S2P <- diff_px_citel(trzby_citel_new_S2P)
```

```
diff_trzby_citel_new_S2Q <- diff_px_citel(trzby_citel_new_S2Q)
```

```
diff_trzby_citel_new_S3P <- diff_px_citel(trzby_citel_new_S3P)
```

```
diff_trzby_citel_new_S3Q <- diff_px_citel(trzby_citel_new_S3Q)
```

```
diff_trzby_citel_new_S4P <- diff_px_citel(trzby_citel_new_S4P)
```

```
diff_trzby_citel_new_S4Q <- diff_px_citel(trzby_citel_new_S4Q)
```

```
diff_trzby_citel_new_S5P <- diff_px_citel(trzby_citel_new_S5P)
```

```
diff_trzby_citel_new_S5Q <- diff_px_citel(trzby_citel_new_S5Q)
```

```
diff_trzby_citel_new_S1Epd <- diff_px_citel(trzby_citel_new_S1Epd)
diff_trzby_citel_new_S2Epd <- diff_px_citel(trzby_citel_new_S2Epd)
diff_trzby_citel_new_S3Epd <- diff_px_citel(trzby_citel_new_S3Epd)
diff_trzby_citel_new_S4Epd <- diff_px_citel(trzby_citel_new_S4Epd)
diff_trzby_citel_new_S5Epd <- diff_px_citel(trzby_citel_new_S5Epd)
```

```
#graf
```

```
output$citel_plot <- renderPlot({
```

```
  options(scipen = 999)
```

```
  fmt_citel <- function(x) format(x, big.mark = ",", scientific = FALSE)
  msg_citel <- paste("Výchozí hodnoty: S1Epd = ", fmt_citel(S1Epd),
                    ", S2Epd = ", fmt_citel(S2Epd),
                    ", S3Epd = ", fmt_citel(S3Epd),
                    ", S4Epd = ", fmt_citel(S4Epd),
                    ", S5Epd = ", fmt_citel(S5Epd))
```

```
  plot(100 * changes_citel, diff_trzby_citel_new_S1Epd,
       xlab = "Změna vstupní proměnné (%)",
       ylab = "Změna tržeb (CZK)",
       main = "Citlivostní analýza",
       type = "l",
       sub = msg_citel)
```

```
  lines(100 * changes_citel, diff_trzby_citel_new_S2Epd, col = "red")
  lines(100 * changes_citel, diff_trzby_citel_new_S3Epd, col = "blue")
  lines(100 * changes_citel, diff_trzby_citel_new_S4Epd, col = "green")
  lines(100 * changes_citel, diff_trzby_citel_new_S5Epd, col = "gray")
```

```
#legenda grafu
```

```
  leg_names_citel <- c("S1Epd", "S2Epd", "S3Epd", "S4Epd", "S5Epd")
  legend("bottomright",
        legend = leg_names_citel,
        col = c("black", "red", "blue", "green", "gray"),
        pch = 15,
        bty = "n")
```

```
})
```



```
}
```

```
citl_analyza_citel()
```

```
})
```

```
#action button - simulace overbookingu (ZATÍM NEFUNGUJE! - negenerují se data do  
tabulky, nevím proč; pro potřeby knihy budu kopírovat samostatný graf ze skriptu  
"Overbooking-simulace MC.r")
```

```
rvpostob <- reactiveValues(data = NULL)  
observeEvent(input$POSTOBrun, {
```

```
#výpočet
```

```
POSTOBcap <- input$POSTOBcap  
POSTOBprshow <- input$POSTOBprshow  
POSTOBp <- input$POSTOBp  
POSTOBcsteh <- input$POSTOBcsteh  
POSTOBobmax <- input$POSTOBobmax
```

```
generate.show <- function(POSTOBoversold, POSTOBprshow) {  
  showup <-  
    sample(c(TRUE, FALSE),  
          POSTOBcap + POSTOBoversold,  
          replace = TRUE,  
          prob = c(POSTOBprshow, 1-POSTOBprshow))  
  sum(showup)  
}
```

```
resultPOSTOB <- data.frame(  
  POSTOBoversold = NA,  
  POSTOBprofit = NA  
)
```

```
POSTOBsim <- 1000
```

```
for (POSTOBoversold in 0:POSTOBobmax) {  
  for (i in 1:POSTOBsim) {  
    showup_rand <- generate.show(POSTOBoversold, POSTOBprshow)  
    POSTOBTR = showup_rand * POSTOBp  
    comp_total <- ifelse (showup_rand >= POSTOBcap,
```

```

        (showup_rand - POSTOBcap) * POSTOBcsteh,
        0)
    POSTOBprofit <- POSTOBTR - comp_total
    resultPOSTOB <- rbind(resultPOSTOB, c(POSTOBoversold, POSTOBprofit))
  }
}

```

```
# graf
```

```
output$POSTOBplot <- renderPlot({
```

```

  ggplot(
    resultPOSTOB %>%
      filter(!is.na(POSTOBoversold)),
    aes(x = factor(POSTOBoversold), y = POSTOBprofit)
  ) +
  geom_boxplot()+
  labs(#title = "Possible Total Revenue for Tickets Oversold",
       x = "Počet pokojů rezervovaných nad kapacitu",
       y = "Očekávaný zisk") +
  theme(legend.position = "none")
})

```

```
})
```

```
### 8. panel
```

```

#action button
rvp <- reactiveValues(data = NULL)
observeEvent(input$Prun, {

  #výpočet P (základní nabídkové)
  opakujP <- replicate (n=200, {
    x0 <- c(1)
    fn <- function (x) -((input$PQD + rnorm(n=100, input$PEPD, 0.05) * (x -
input$PLRV) * (input$PQD/input$PLRV)) * x)
    hin = function (x) {
      h <- numeric (2)
      h[1] <- x - input$Pcost
    }
  })
}

```

```

    h[2] <- x
    return(h)
  }

  res <- auglag (x0, fn, gr = NULL, hin = hin)
  solution <- res$par
}, simplify = F)

nabcena <- as.numeric(opakujP)

#graf
output$Pplot <- renderPlot({
  hist(nabcena, main = ("Základní nabídková cena"), breaks = 30, col = "grey", freq =
F)
  curve(dnorm(x, mean = mean(nabcena), sd = sd(nabcena)), add = T, col = "red")
})

#hodnoty a doporučení
output$Phodnoty <- renderPrint({
  print(summary(nabcena))
})

#o kolik procent je doporučení na změnu
output$Pdoporuceni <- renderText({
  (mean(nabcena) / input$PLRV * 100) > 100
})
})

#action button - cena konkrétní rezervace
rvpb <- reactiveValues(data = NULL)
observeEvent(input$Pbrun, {
  #výpočty

  VPznc <- input$Pznc
  VMUp <- (input$Pbmax - input$Pbmin) * input$Pocc + input$Pbmin - input$Pznc
  VPRb <- input$Pznc - ((input$Pfc + input$Pavc * input$Pocc * input$Pcap)/(input$Pocc
* input$Pcap))
  VPIb <- input$PIb
  VPtheta <- input$Ptheta

  if (VPRb < 0) {VPRb = 0}
  else {VPRb <- VPRb}

  Prez <- VPznc + VMUp - VPRb + VPIb + VPtheta

```

```

Prezw <- VPznc + VMUp + VPIb + VPtheta

if (Prez > input$Pbmax) {Prez <- input$Pbmax}
else {Prez <- Prez}

if (Prez < input$Pbmin) {Prez <- input$Pbmin}
else {Prez <- Prez}

if (Prezw > input$Pbmax) {Prezw <- input$Pbmax}
else {Prezw <- Prezw}

if (Prezw < input$Pbmin) {Prezw <- input$Pbmin}
else {Prezw <- Prezw}

#výstupy (ukázání výsledku)
output$Pboutput <- renderText({
  Prez
})

output$Pboutputw <- renderText({
  Prezw
})

output$Pbpozor <- renderText({
  if (input$Pbmin < input$Pavc) {"Pozor - hodnota variabilních nákladů je vyšší než
úroveň mininální ceny"}
  else {"Žádné upozornění"}
})

})

#### 9. panel (odhad FC)

#vstupní hodnoty nahrané ze souboru (nejsou součástí action button)
output$vsstupyFC <- renderDataTable({
  if(is.null(input$souborFC)){return()}
  input$souborFC
  dotabulkyFC <- input$souborFC
  aaFC <- read_xls(dotabulkyFC$datapath)
})

#action button
rvregFC <- reactiveValues(data = NULL)

```

```

observeEvent(input$regreseFC, {

  #výpočty
  dataRFC <- read_xls(input$souborFC$datapath)
  R <- dataRFC$R
  TC <- dataRFC$TC
  lregreseFC <- lm(TC ~ R)
  hodnotyRFC <- summary(lregreseFC)
  vysledkyRFC <- as.numeric(lregreseFC$coefficients)
  FC <- vysledkyRFC[c(1)]

  #výstup data v tabulce
  output$resultFC <- renderPrint({
    hodnotyRFC
  })

  #výstup graf
  output$regreseFCplot <- renderPlot({
    plot(TC ~ R, xlim = c(0, max(R)), ylim = c(0, max(TC)))
    abline(reg = lregreseFC)
  })

  #výstup hodnota FC
  output$resFC <- renderPrint({
    FC
  })

})

### 10. panel
#action button
rvavc <- reactiveValues(data = NULL)
observeEvent(input$AVCrun, {

  #výpočty
  AVC <- input$AVCnarez + input$AVCuklid + input$AVCFaB + input$AVCudrzba +
input$AVCmat + input$AVCmzda + input$AVCenerg + input$AVCcost

  #výstup
  output$AVCoutput <- renderText({
    AVC
  })

})

### 11. panel

```

```
#actionbutton
rvpmin <- reactiveValues(data = NULL)
observeEvent(input$Pminrun, {

  #výpočty
  Pmin <- ((input$PminBL/input$Pmincap) * input$PminFC + input$PminAVC *
input$Pminocc * input$PminBL) / (input$Pminocc * input$PminBL * (1-input$PminROS))

  #výstup
  output$Pminres <- renderText({
    Pmin
  })
})

}
```

```
shinyApp(ui = ui, server = server)
```

(c) Martin Petricek (draft version of the code) - STORM (STOchastic Revenue Management model)