The background of the slide is a light gray gradient with several realistic water droplets of various sizes scattered across it. The droplets have highlights and shadows, giving them a three-dimensional appearance.

SHINY를 이용한 MOBILE APP 만들기

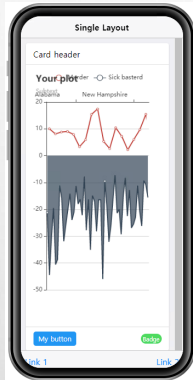
- SHINYMOBILE 패키지 소개 -

NAM-HYEON JEE

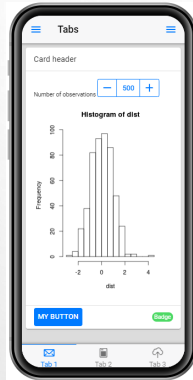
1. shinyMobile 소개

- [shinyMobile](#)은 [framework7](#) 기반의 모바일용 샤이니앱 개발 패키지
- 3가지 기본 레이아웃 (SingleLayout, TabLayout, SplitLayout) 제공

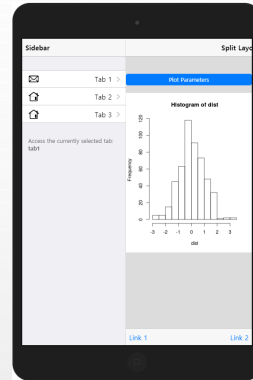
SingleLayout



TabLayout



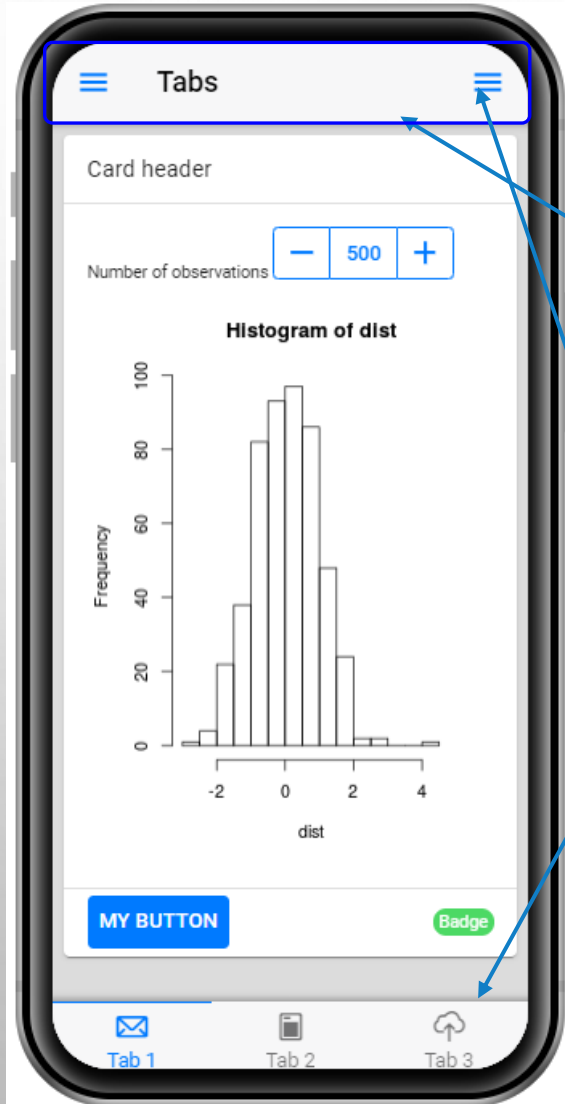
SplitLayout



- 3가지 스킨 (모바일용 ios 및 md, 데스크톱용 aurora) 제공
- 2가지 테마 (light, dark) 제공
- 각 스킨에 대한 고유한 디자인을 가진 자체 위젯 제공

※ [RinteRface](#) : shinydashboard+, bs4Dash, shinyEffects 등 개발

2. 레이아웃



TabLayout

- **Page (f7Page)**
shinyMobile 레이아웃을 정의하는 부분
- **Navbar (f7Navbar)**
앱 이름, 부제목, 패널 사용 여부 등 표시
- **Toolbar (f7Toolbar) 또는 Tab (f7Tabs)**
메인 메뉴 기능
- **Panel (f7Panel)**
좌우 사이드바 메뉴 구성

```
ui = f7Page(  
  title = "Tabs",  
  init = f7Init(skin = "ios"),  
  
  f7STabLayout(.  
    ...,  
    navbar,  
    toolbar,  
    panels  
  )  
)
```

1) SingleLayout

간단한 한 페이지로 구성된
모바일용 앱 개발

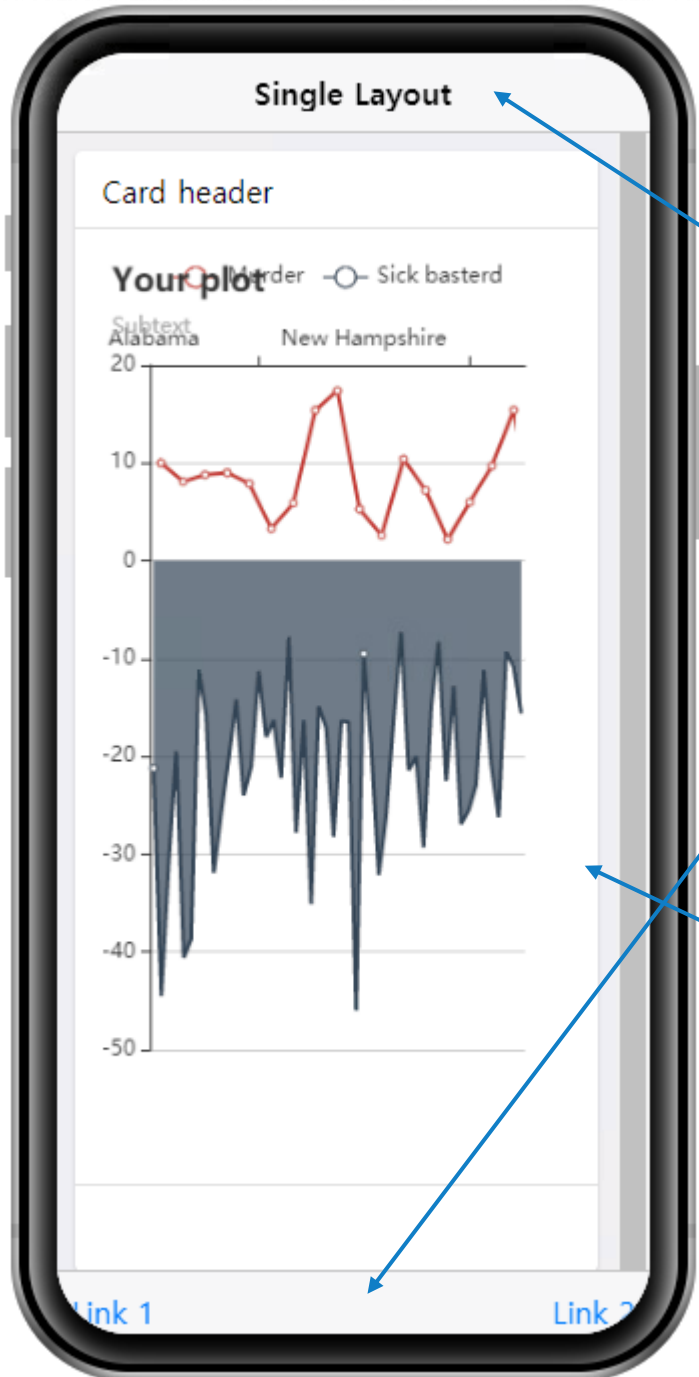
```
ui = f7Page(  
  title = "My app",  
  dark_mode = FALSE,  
  init = f7Init(skin = "ios", theme = "light"),
```

```
f7SingleLayout(  
  navbar = f7Navbar(  
    title = "Single Layout",  
    hairline = TRUE,  
    shadow = TRUE  
  ),  
  toolbar = f7Toolbar(  
    position = "bottom",  
    f7Link(label = "Link 1", src = "https://www.google.com"),  
    f7Link(label = "Link 2", src = "https://www.google.com", external = TRUE)  
  ),  
  f7Shadow(  
    intensity = 16,  
    hover = TRUE,  
    f7Card(  
      title = "Card header",  
      apexchartOutput("areaChart")  
    )  
  )  
)
```

```
  navbar = f7Navbar(  
    title = "Single Layout",  
    hairline = TRUE,  
    shadow = TRUE  
  ),
```

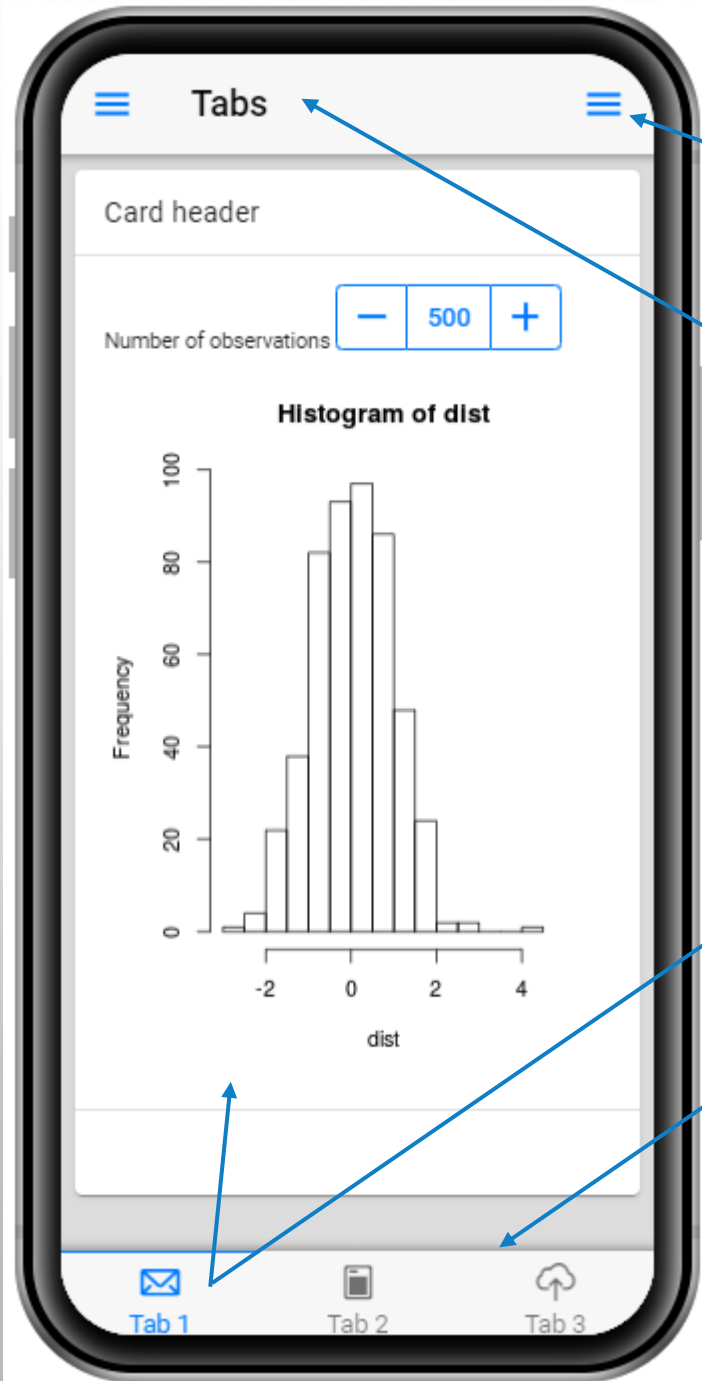
```
  toolbar = f7Toolbar(  
    position = "bottom",  
    f7Link(label = "Link 1", src = "https://www.google.com"),  
    f7Link(label = "Link 2", src = "https://www.google.com", external = TRUE)  
  ),
```

```
  f7Shadow(  
    intensity = 16,  
    hover = TRUE,  
    f7Card(  
      title = "Card header",  
      apexchartOutput("areaChart")  
    )  
  )  
)
```



2) TabLayout

패널 및 다중 탭으로 구성된 모바일용 앱 개발



```
ui = f7Page(  
  title = "My app",  
  init = f7Init(theme = "dark"),  
  f7TabLayout(  
    panels = tagList(  
      f7Panel(title = "Left Panel", side = "left", theme = "light"),  
      f7Panel(title = "Right Panel", side = "right", theme = "dark")  
    ),  
    navbar = f7Navbar(  
      title = "Tabs",  
      hairline = TRUE,  
      shadow = TRUE,  
      left_panel = TRUE,  
      right_panel = TRUE  
    ),  
    f7Tabs(  
      animated = TRUE,  
      f7Tab(  
        tabName = "Tab 1",  
        icon = f7Icon("email"),  
        active = TRUE,  
  
        f7Shadow(  
          intensity = 10,  
          hover = TRUE,  
          f7Card(  
            title = "Card header",  
            plotOutput("hist")  
          )  
        )  
      )  
    )  
  )  
)
```

```
panels = tagList(  
  f7Panel(title = "Left Panel", side = "left", theme = "light"),  
  f7Panel(title = "Right Panel", side = "right", theme = "dark")  
)
```

```
navbar = f7Navbar(  
  title = "Tabs",  
  hairline = TRUE,  
  shadow = TRUE,  
  left_panel = TRUE,  
  right_panel = TRUE  
)
```

```
f7Tabs(  
  animated = TRUE,  
  f7Tab(  
    tabName = "Tab 1",  
    icon = f7Icon("email"),  
    active = TRUE,  
  
    f7Shadow(  
      intensity = 10,  
      hover = TRUE,  
      f7Card(  
        title = "Card header",  
        plotOutput("hist")  
      )  
    )  
  )  
)
```

```
f7Tab(  
  tabName = "Tab 2",  
  ...  
)  
f7Tab(  
  tabName = "Tab 3",  
  ...  
)  
)  
)  
)
```

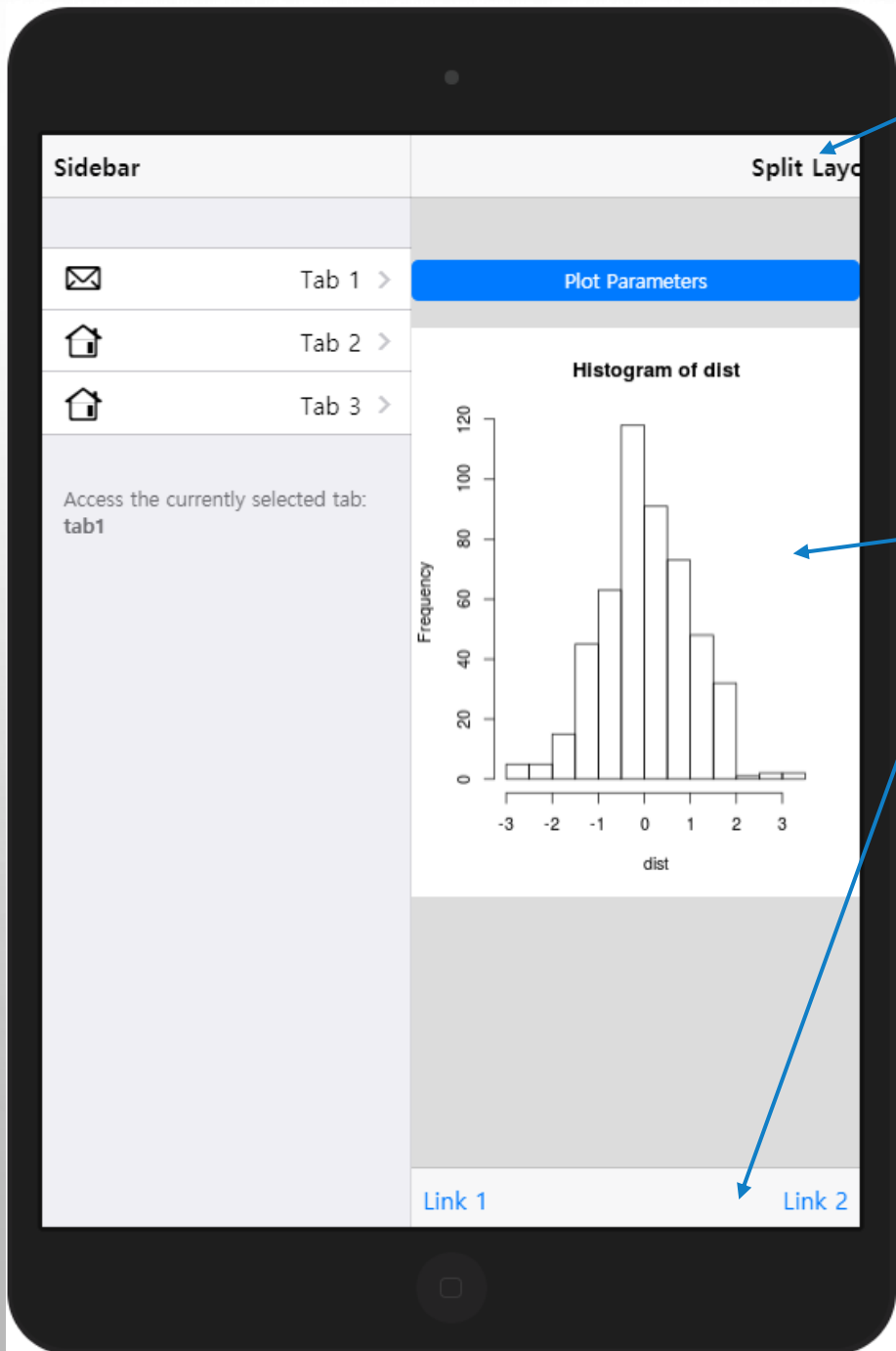
3) SplitLayout

사이드바 및 메인 패널로 구성된

태블릿 및 데스크톱용 앱 개발



```
ui = f7Page(  
  title = "My app",  
  init = f7Init(  
    hideNavOnPageScroll = FALSE,  
    hideTabsOnPageScroll = FALSE,  
    theme = "light"  
  ),  
  f7SplitLayout(  
    sidebar = f7Panel(  
      title = "Sidebar",  
      side = "left",  
      theme = "light",  
      f7PanelMenu(  
        id = "menu",  
        f7PanelItem(  
          tabName = "tab1",  
          title = "Tab 1",  
          icon = f7Icon("email"),  
          active = TRUE  
        ),  
        f7PanelItem(  
          tabName = "tab2",  
          ...  
        ),  
        f7PanelItem(  
          tabName = "tab3",  
          ...  
        )  
      )  
    ),  
  ),  
),
```



```

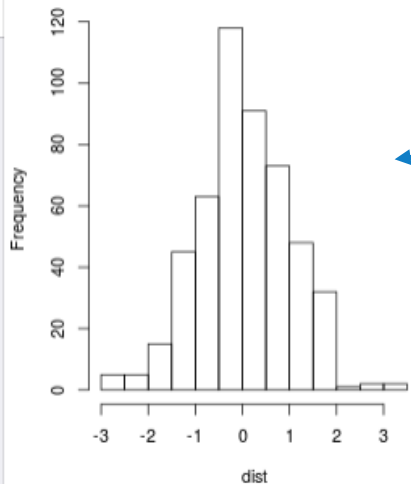
navbar = f7Navbar(
  title = "Split Layout",
  hairline = FALSE,
  shadow = TRUE
),
toolbar = f7Toolbar(
  position = "bottom",
  f7Link(label = "Link 1", src = "https://www.google.com"),
  f7Link(label = "Link 2", src = "https://www.google.com")
),
# main content
f7Items(
  f7Item(
    tabName = "tab1",
    plotOutput("distPlot")
  ),
  f7Item(
    tabName = "tab2",
    ...
  ),
  f7Item(
    tabName = "tab3",
    ...
  )
)
)
)

```

Split Layo

Plot Parameters

Histogram of dist



Link 1

Link 2


3. 데모

- 1) Single Layout : <https://rinterface.github.io/shinyMobile/articles/Single-Layout.html>
- 2) Tabs Layout : <https://rinterface.github.io/shinyMobile/articles/Tabs-Layout.html>
- 3) Split Layout : <https://rinterface.github.io/shinyMobile/articles/Split-Layout.html>

4. Cheat Sheet

- shinyMobile Reference : <https://rinterface.github.io/shinyMobile/reference/index.html>
- Cheat Sheet : <https://github.com/RinterFace/cheatsheets/blob/master/shinyMobile/shinyMobile.pdf>

shinyMobile: : CHEAT SHEET



Basics

shinyMobile is a template to design mobile ready and desktop Shiny Apps. shinyMobile is leveraging the progressive web app capability, thereby providing a standalone/fullscreen support.

```
install.packages("shinyMobile")
Remotes::install_github("RinterFace/shinyMobile")
```

You can setup custom desktop icons and splashscreen to produce a native app feeling

3 TEMPLATES

shinyMobile has 3 predefined templates:

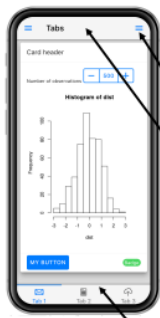
- f7SingleLayout(..., navbar, toolbar = NULL, panels = NULL, appbar = NULL, statusbar = f7Statusbar())**: one page layout
- f7TabLayout(..., navbar, panels = NULL, appbar = NULL, statusbar = f7Statusbar())**: tab layout
- f7SplitLayout(..., navbar, sidebar, toolbar = NULL, panels = NULL, appbar = NULL, statusbar = f7Statusbar())**: ipad specific layout with sidebar panel and main panel

f7Page is the main wrapper function

```
library(shiny)
library(shinyMobile)
ui <- f7Page()
server <- function(input, output){
  shinyApp(ui = ui, server = server)
}
```

Layout Examples: Simple and Tabs (1)

TAB LAYOUT

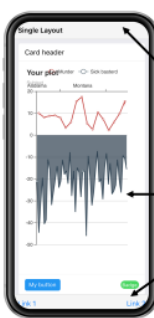


```
library(shiny)
library(shinyMobile)

shiny::shinyApp(
  ui = f7Page(
    title = "Tab Layout",
    f7TabLayout(
      # panels are not mandatory. These are similar to sidebars
      panels = tagList(
        f7Panel(side = "left", theme = "light", effect = "cover", ...),
        f7Panel(side = "right", theme = "dark", effect = "reveal", ...)
      ),
      navbar = f7Navbar(
        title = "Tabs",
        # enable both panels
        left_panel = TRUE,
        right_panel = TRUE
      ),
      # f7Tabs is a special toolbar with included navigation
      f7Tabs(
        animated = TRUE,
        id = "tabs",
        f7Tab(
          tabName = "Tab 1",
          icon = f7Icon("email"),
          active = TRUE,
          # Tab 1 content
          ...
        ),
        # Other tabs
        ...
      )
    ),
    in = input$tab
  )
  server = function(input, output) {}
)
```

f7TabLayout is probably the most versatile template since it adapts to many situations. By specifying an id to **f7Tabs**, you may recover the currently selected tab in `input$tab`

SINGLE LAYOUT

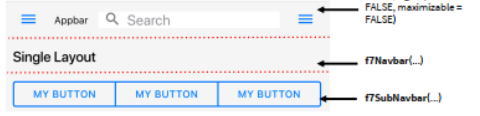


```
library(shiny)
library(shinyMobile)

ui <- f7Page(
  title = "My app",
  f7SingleLayout(
    # navbar is mandatory
    navbar = f7Navbar(
      title = "Single Layout",
      hairline = FALSE,
      shadow = TRUE
    ),
    # main content
    ...
    # toolbar
    toolbar = f7Toolbar(
      position = "bottom",
      # toolbar content
      ...
    )
  )
  server <- function(input, output){
    shinyApp(ui = ui, server = server)
  }
)
```

f7SingleLayout is well suited for simple prototype apps, with only one view.

COMMON LAYOUT ELEMENTS



```
f7Appbar(..., left_panel = FALSE, right_panel = FALSE, maximizable = FALSE)
f7Navbar(...)
f7SubNavbar(...)
```

f7TabLayout and **f7SingleLayout** have an extra slot for an appbar. **f7Appbar** is an extra navigation bar!

ONLY FOR SPLIT LAYOUT


```
f7PanelMenu(..., id = NULL)
f7PanelItem(title, tabName, icon = NULL, active = FALSE)
```

When used in a **split layout** context, `input$<inputid>` returns the currently select panel item. These items may be passed in the ...

BOOKMARK YOUR APP:

Host it on shinyapps.io, RStudio Connect (www.rstudio.com/products/connect/) or Build your own Shiny Server (www.rstudio.com/products/shiny-server/)

- Visit your app using your favorite web browser (Chrome, Firefox, ...)
- Depending on the web browser, select the add to home screen feature
- Change the name
- Click on ok
- Enjoy!



RinterFace-<https://rinterface.com> • David Granjon • dgranjon@gmail.com • <https://github.com/dgranjon> • Learn more at <https://rinterface.github.io/shinyMobile/> • package version 0.2.0 • Updated: 2020-01

5. 모바일 웹앱 제작

- SWING2APP  SWING2APP

- 무료 모바일 앱 제작 사이트

- 회원 가입 ▶ 앱 제작 (http://www.swing2app.co.kr/maker_view/basic_info)

- ▶ 디자인 테마 : 웹뷰 전용



- 앱 제작이 완료되면, 가입 이메일로 다운로드 링크 전달됨

모든 단계가 완료되었습니다, 앱 제작 요청을 진행하시겠습니까?

확인

취소

6. 무료 개발 서버

- K-ICT 빅데이터 센터 
 - 분석·시각화 지원 소프트웨어, 서버 등 인프라 지원 및 오픈랩 제공
 - 회원 가입 ▶ 인프라 예약 (<https://kbig.kr/portal/kbig/analysis/infra/reserve.page>)
 - 자원 : 8 Core, 16G RAM, 150G HDD, CentOS 7
 - 포트할당 요청 필요

* 개발 내용	요청사항	SSH 접속 정보	IP : 210.114.91.91 Port : 26034	* Hard	150 Gb
	1) 포트 할당 요청			* Cpu	8 Core
	Port1 : 3838	서비스포트1	=> 26035	* Memory	16 Gb
	Port2 : 8787	서비스포트2	=> 26036	* OS	CentOS 7.0
	Port3 : 54321	서비스포트3	=> 26037	* Web	Apache2 2.4
	2) devuser에 root 권한 추가	아이디	devuser	* DB	PostgreSQL
		비밀번호	hello.dev		

※ 딥러닝 서버는 tensorflow 1.x 이며, 업데이트 불가함 (추천 x)

7. Shiny App 개발에 유용한 링크

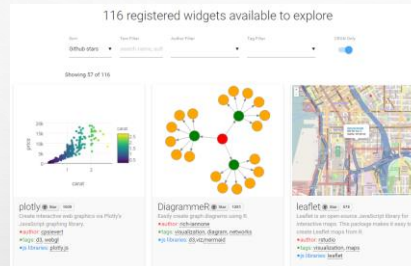
1) Awesome Shiny Extensions

- 카테고리별로 다양한 Shiny 확장 기능 소개
- <https://github.com/nanxstats/awesome-shiny-extensions>



2) htmlwidgets for R

- 100여개의 interactive chart 소개
- <http://gallery.htmlwidgets.org/>



3) Mastering Shiny

- <https://mastering-shiny.org/>

