

MENDELU 2023

Tomas Cejka(CJ), Róbert Oravec - STRV

STRV

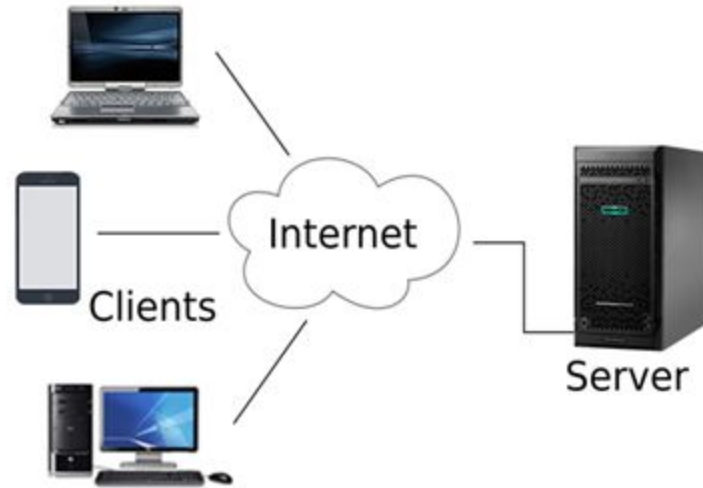
STRV X MENDELU

- History
- Lecture + seminars
- Autumn iOS vol.2
- Networking
- Questions
- 🖐️

NETWORKING

STRV

NETWORKING



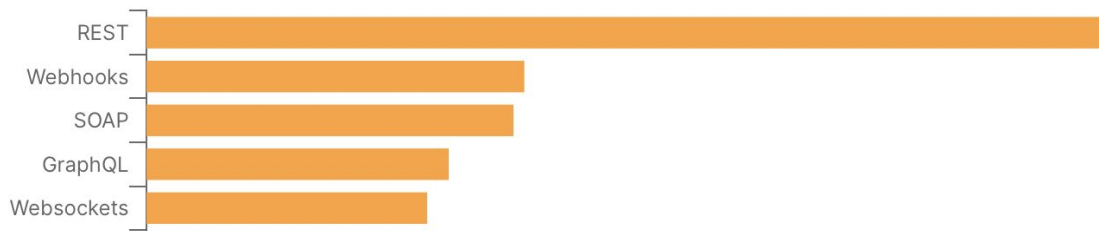
NETWORKING

- Async
- Remote server
- Authentication
- Request creation
- Error handling
- Parsing data



NETWORKING

- REST - 89 %
- Webhooks - 35 %
- SOAP - 34 %
- GraphQL - 28 %
- Websockets - 26 %



REST



01

WHAT IS REST ?

- API - Application Programming Interface
- REST - REpresentational State Transfer
- Architectural style and approach to communications often used in web services development



REST

- Uniform interface
- Client - server
- Statelessness
- Layered system



ANATOMY OF URL

`https://rickandmortyapi.com/api/character?page=20`

The diagram illustrates the components of the URL `https://rickandmortyapi.com/api/character?page=20`. The text is color-coded: `https://` is black, `rickandmortyapi.com` is red, `/api/character` is orange, and `?page=20` is blue. Below the URL, three labels are positioned with vertical lines pointing to their respective parts: `HOST` (red) points to `rickandmortyapi.com`, `PATH` (orange) points to `/api/character`, and `QUERY` (blue) points to `?page=20`.

HOST

PATH

QUERY

HTTP METHODS

- GET
- POST
- PUT
- PATCH
- DELETE

HTTP HEADERS

- Content-type
- Accept
- Content-Encoding
- Cache-control
- Authorization

```
GET /home.html HTTP/1.1
Host: developer.mozilla.org
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:50.0) Gecko/20100101
Firefox/50.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://developer.mozilla.org/testpage.html
Connection: keep-alive
Upgrade-Insecure-Requests: 1
If-Modified-Since: Mon, 18 Jul 2016 02:36:04 GMT
If-None-Match: "c561c68d0ba92bbeb8b0fff2a9199f722e3a621a"
Cache-Control: max-age=0
```

HTTP BODY

- Application/x-www-form-urlencoded
- Multipart/form-data
- Text/plain
- Application/json

```
HTTP/1.1 200 OK
Date: Sun, 10 Oct 2010 23:26:07 GMT
Server: Apache/2.2.8 (Ubuntu) mod_ssl/2.2.8 OpenSSL/0.9.8g
Last-Modified: Sun, 26 Sep 2010 22:04:35 GMT
ETag: "45b6-834-49130cc1182c0"
Accept-Ranges: bytes
Content-Length: 12
Connection: close
Content-Type: text/html

Hello world!
```

HTTP RESPONSE STATUS CODES

- Information responses (100 - 199)
- Successful responses (200 - 299)
 - 200 - everything went ok
 - 201 - resource created using for example POST or PUT method
- Redirection responses (300 - 399)
- Client error responses (400 - 499)
 - 400 - bad request
 - 401 - unauthorized
 - 403 - forbidden
 - 404 - not found
- Server error responses (500 - 599)

HTTP RESPONSE TYPES

- XML
- JSON
- YAML
- HTML
- Binary

JSON

- Data interchange object
- Simple format
 - Constraints
 - Readable for human
 - Basic data types
 - Small

JSON VS XML

JSON Example

```
{ "employees": [
  { "firstName": "John", "lastName": "Doe" },
  { "firstName": "Anna", "lastName": "Smith" },
  { "firstName": "Peter", "lastName": "Jones" }
]}
```

XML Example

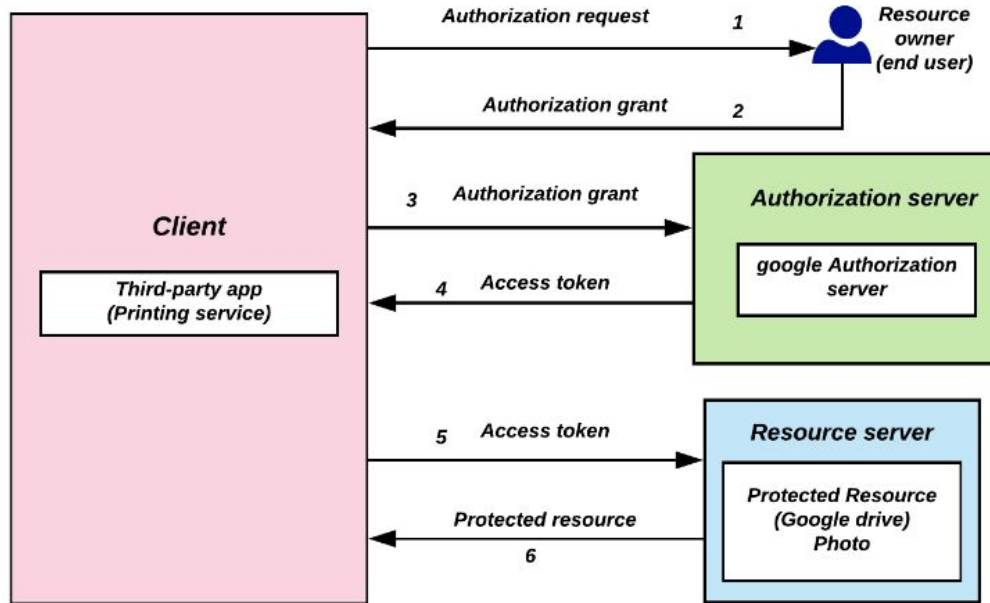
```
<employees>
  <employee>
    <firstName>John</firstName> <lastName>Doe</lastName>
  </employee>
  <employee>
    <firstName>Anna</firstName> <lastName>Smith</lastName>
  </employee>
  <employee>
    <firstName>Peter</firstName> <lastName>Jones</lastName>
  </employee>
</employees>
```

AUTHENTICATION

- Basis // Authorization: Basic bG9sOnNlY3VyZQ==
- Bearer // Authorization: Bearer <token>
- API key // Authorization: Apikey 1234567890abcdef
- OAuth
 - Authorization server
 - Access token + Refresh token
 - Facebook, Google, Apple sign in
- JWT, OpenID

OAUTH

Abstract Flow





QUESTION?

STRV

EXAMPLES



02

REST + IOS



03

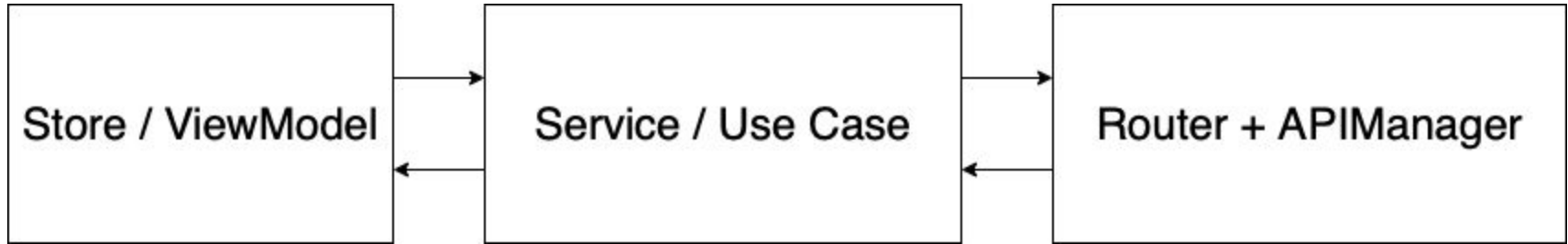


URLSESSION

- Downloading data from and uploading data to endpoints
- Singleton instance can be accessed through shared static property
 - Default
 - Ephemeral session - no cache
 - Background session - when app is not running
- Data tasks - short requests
- Upload tasks - support background session
- Download tasks - support background session
- WebSocket tasks
- Alamofire + Moya



NETWORK LAYERS IN IOS APP



API MANAGER

- uses URLSession to call finalized network request
- authentication, automatic refresh token
- wrong status code responses

```
protocol APIManaging {
    @discardableResult
    func dataTask(from request: URLRequest) async throws -> Data
}

extension APIManaging {
    func modelTask<T>(
        from request: URLRequest,
        decoder: JSONDecoder = JSONDecoder()
    ) async throws -> T where T: Decodable {
        try decoder.decode(T.self, from: try await dataTask(from: request))
    }
}
```

ROUTER

- creates URLRequest

```
protocol Rounter {  
    var host: String { get }  
    var path: String { get }  
    var query: Query { get }  
    var method: HTTPMethod { get }  
    var headers: HTTPHeaders { get }  
    var body: HTTPBody? { get }  
  
    func createURLRequest() throws -> URLRequest  
}
```

```
enum CharacterRouter: Rounter {  
    case characters(page: Int)  
  
    var path: String {  
        switch self {  
            case .characters:  
                return "/characters"  
        }  
    }  
  
    var query: Query {  
        switch self {  
            case let .characters(page):  
                return ["page": String(page)]  
        }  
    }  
}
```

USE CASE / SERVICE

- Creates request using Router and then call it with APIManager

```
final class CharacterService {  
    private let apiManager: APIManaging  
  
    init(apiManager: APIManaging) {  
        self.apiManager = apiManager  
    }  
  
    func getCharacters(at page: Int = 0) async throws -> CharacterResults {  
        let request = try CharacterRouter.characters(page: page)  
            .createURLRequest()  
  
        return try await apiManager.modelTask(from: request)  
    }  
}
```

CODABLE

- Protocol used to convert itself to and out of external representation (for example JSON)
- Codable = Encodable & Decodable
- Introduced in Swift 4
- JSONEncoder, JSONDecoder
- CodingKey

```
{  
    "id": 361,  
    "name": "Toxic Rick",  
    "status": "Dead",  
    "species": "Humanoid",  
    "type": "Rick's Toxic Side",  
    "gender": "Male"  
}
```

```
struct Character: Decodable {  
    let id: Int  
    let name: String  
    let status: String  
    let species: String  
    let type: String  
    let gender: String  
}
```

REST PLAYGROUND

04

QUESTIONS



STRV

THANK YOU!

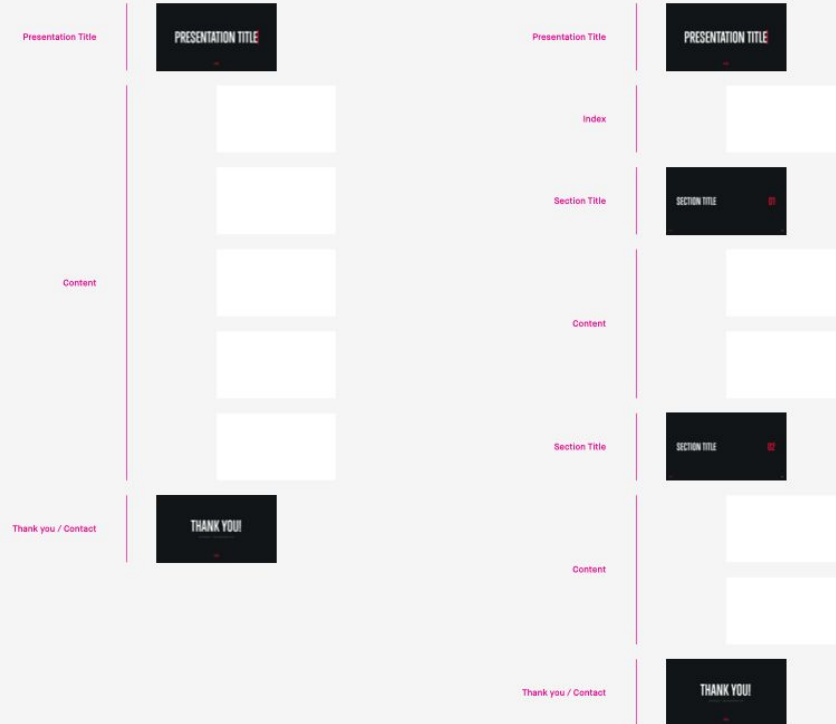
www.strv.com / [@strvcom](https://twitter.com/strvcom)

STRV

- 1) Get the official **STRV fonts** at brand.strv.com before creating a presentation
- 2) Keep the presentation structure described on Slide 3
- 2) Edit **content only**, not formatting or layout
- 3) Don't over-do it with GIFs
- 4) Send your finished presentation to the Design team for a **review**
- 5) Presentations without a review should not be shared publicly

Presentation structure

Level 1 → Level 2



Do not use this slide in your presentation, it's just a section break in this template!

PRESENTATION TITLE

Do not use this slide in your presentation, it's just a section break in this template!

PRESENTATION TITLE

STRV

PRESENTATION TITLE

[Name][Surname], [Position] at STRV

STRV

PRESENTATION TITLE WITH TWO LINES

STRV

PRESENTATION TITLE WITH TWO LINES

[Name][Surname], [Position] at STRV

STRV

SECTION TITLE

Do not use this slide in your presentation, it's just a section break in this template!

SECTION TITLE

01

SECTION TITLE WITH TWO LINES

01

TEXT SLIDES

Do not use this slide in your presentation, it's just a section break in this template!

SLIDE TITLE

- This is the most basic text content slide int, or use a full sentence to highlight something
- Go straight to the point, or use a full sentence to highlight something like this
- Use only one-line texts, beware of splitting text into two rows
- Don't overload text screen too much, 6-8 points are enough
 - You can also use second level
 - Describe more stuff more in detail if necessary

SLIDE TITLE

- Don't let this text overlay your image
- Always choose the image wisely
- Try to not split your point into two rows
- 6-8 points are enough



1 x 1

SLIDE TITLE

75%

Client leads increase

34K

Daily Active Users

3

Total Devs (mo)

SLIDE TITLE

You can also start your content screen with a paragraph before you introduce a couple of bullet points. But stick with 2-3 rows top here so the text isn't too long.

- Now you can use bullet point list
- Points out some highlights
- And use 3-5 points max.

SLIDE TITLE

First section subtitle

- Use this in case you need to have some sections
- Keep section titles short and clear
- 2-3 points are enough

Second section subtitle

- You can also use 2-3 points in this section
- This slide is nicely structured
- With 6 points in total

SLIDE TITLE

First section subtitle

- You place two list next to each other
- Make sure there is not too much text
- It does not look that pretty
- Only use this when necessary
- Otherwise choose a simple option

Second section subtitle

- You place two list next to each other
- Make sure there is not too much text
- It does not look that pretty
- Only use this when necessary
- Otherwise choose a simple option

SLIDE TITLE

- Describe your code here
- I have no idea what this router shit means

```
1  Import { Router, Route } from 'react-router'
2  ReactDOM.render(
3    <Router>
4      <Route component={App} >
5        <Route path="/user" component={User} />
6        <Route path="/user/:id" component={User} />
7      </Route>
8    </Router>
9  )
```

SLIDE TITLE

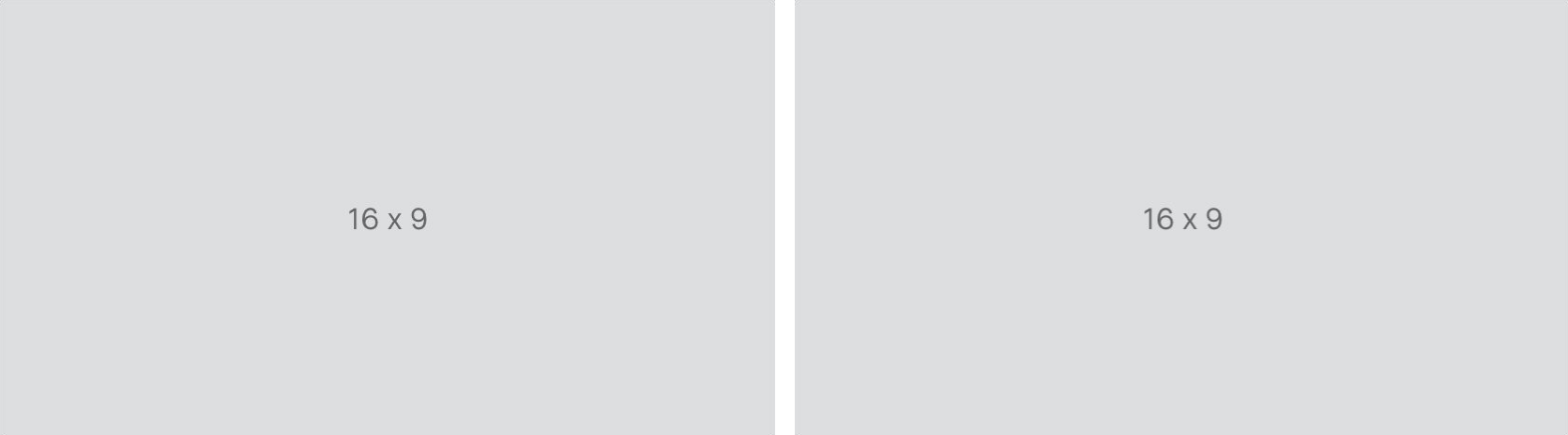
```
1 import React, { Component } from 'react'
2 import { connect } from 'react-redux'
3
4 class LineupList extends Component {
5   render() {
6     return (
7       <h1>LineupList</h1>
8     )
9   }
10 }
11
12 function select(state) {
13   return {
14     lineups: state.lineups
15   }
16 }
17
18 export default connect(select)(LineupList)
```

```
1 import React, { Component } from 'react'
2 import { connect } from 'react-redux'
3
4 @connect(
5   state => ({ lineups: state.lineups })
6 )
7 export default class LineupList extends Component {
8   render() {
9     return (
10       <h1>LineupList</h1>
11     )
12   }
13 }
```

CONTENT SLIDES


Do not use this slide in your presentation, it's just a section break in this template!

16 x 9 / 1 x 1



16 x 9

16 x 9



16 x 9

16 x 9

16 x 9





iPhone X - Black
364 x 790 px



iPhone X - Black
364 x 790 px

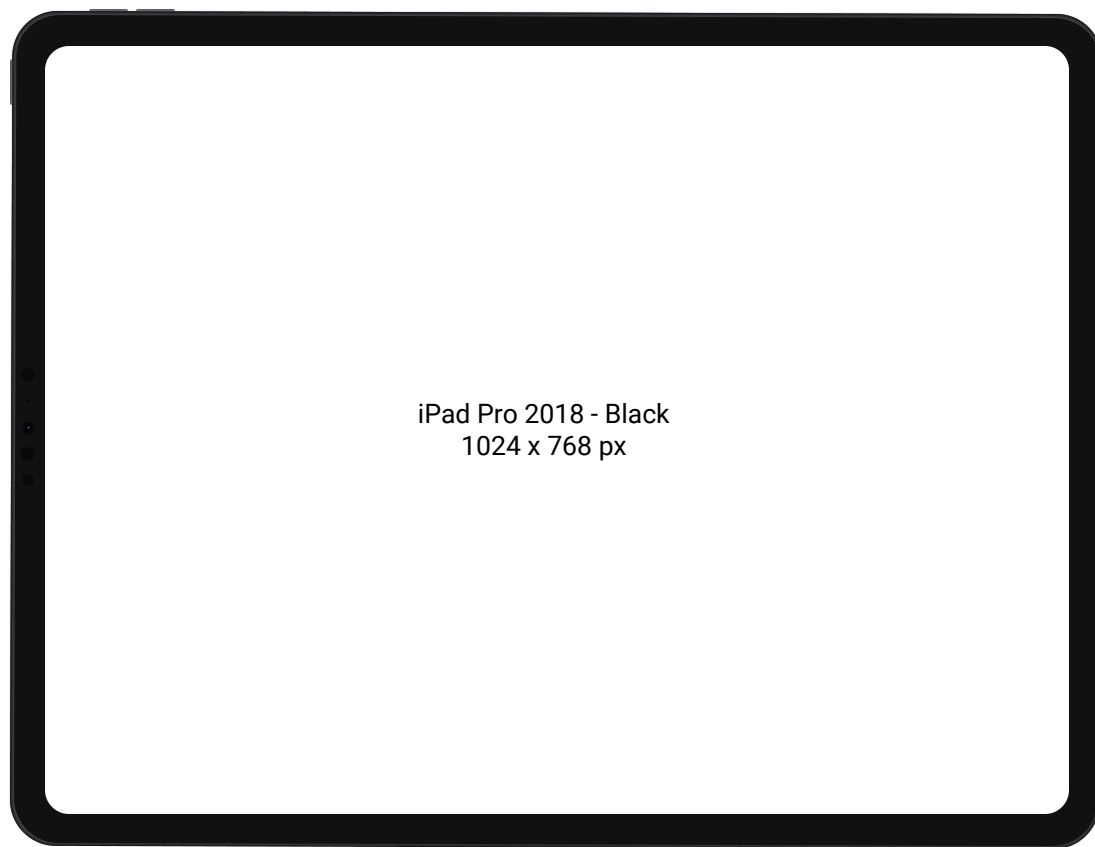




iPhone X - Silver
364 x 790 px

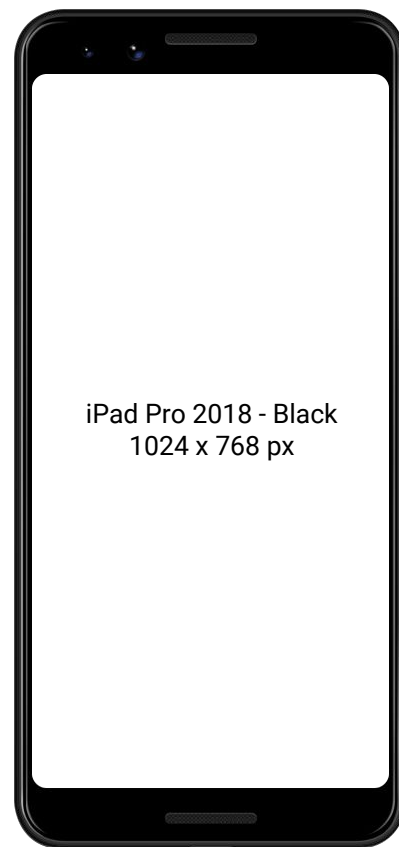


iPhone X - Silver
364 x 790 px

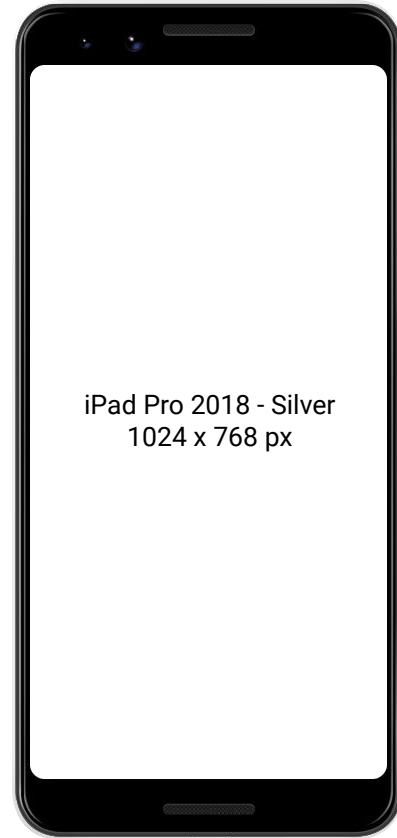












“Creativity is putting your
imagination to work, and it’s
produced the most extraordinary
Results in human culture.”

Ken Robinson, Designer at STRV

FINISH

Do not use this slide in your presentation, it's just a section break in this template!

THANK YOU!

Ales Nesetril / ales.nesetril@strv.com

STRV

ADDITIONAL SLIDES

Do not use this slide in your presentation, it's just a section break in this template!

ABOUT STRV

Design and development of digital solutions for the bravest startups and Fortune 500 companies.

- Software design & engineering team
- Offices in the US and Europe
- More than 190 experts on our side

Visit us at www.strv.com

Follow us at [@strvcom](https://twitter.com/strvcom)



ONE-LINE MESSAGE

ONE-LINE MESSAGE