

# **ITER IDEA's** cloud solutions

What's under the hood - Overview

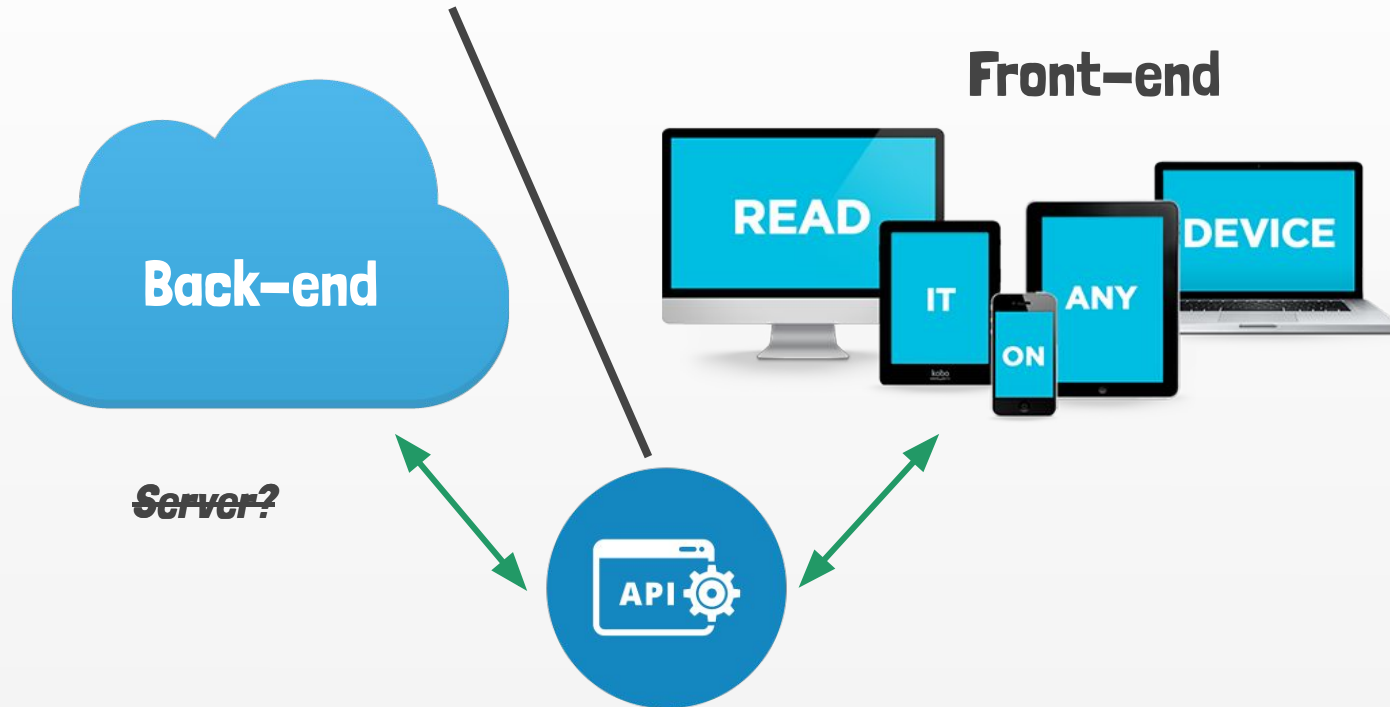
In **ITER IDEA**,

we support companies and organizations developing business **in the cloud**, through applications and serverless solutions: innovative, cost-effective and usable from any device (iOS, Android, Web).

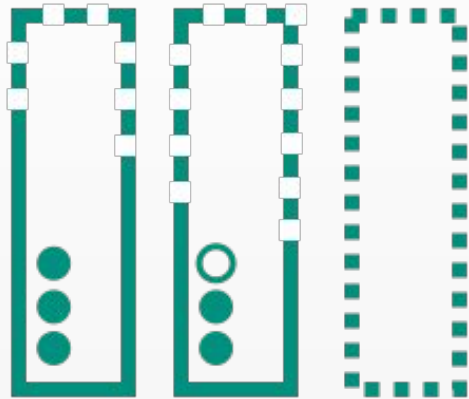


# Architecture, high-level

# Architecture: independent **back-end** and **front-end**



Buzzword: **serverless** cloud



*Back-end*



# Gartner®

## Magic Quadrant for Cloud Infrastructure and Platform Services 2022



- Analytics
- Application Integration
- AR & VR
- AWS Cost Management
- Blockchain
- Business Applications
- Compute
- Containers
- Customer Enablement
- Database
- Developer Tools
- End User Computing
- Front-end Web & Mobile
- Game Development
- Internet of Things
- Machine Learning
- Management & Governance
- Media Services
- Migration & Transfer
- Networking & Content Delivery
- Quantum Technologies
- Robotics
- Satellite
- Security, Identity, & Compliance
- Storage

(Some of the) Amazon Web **Services**;  
*preferred ones: managed, serverless*



## AWS Global Infrastructure Map

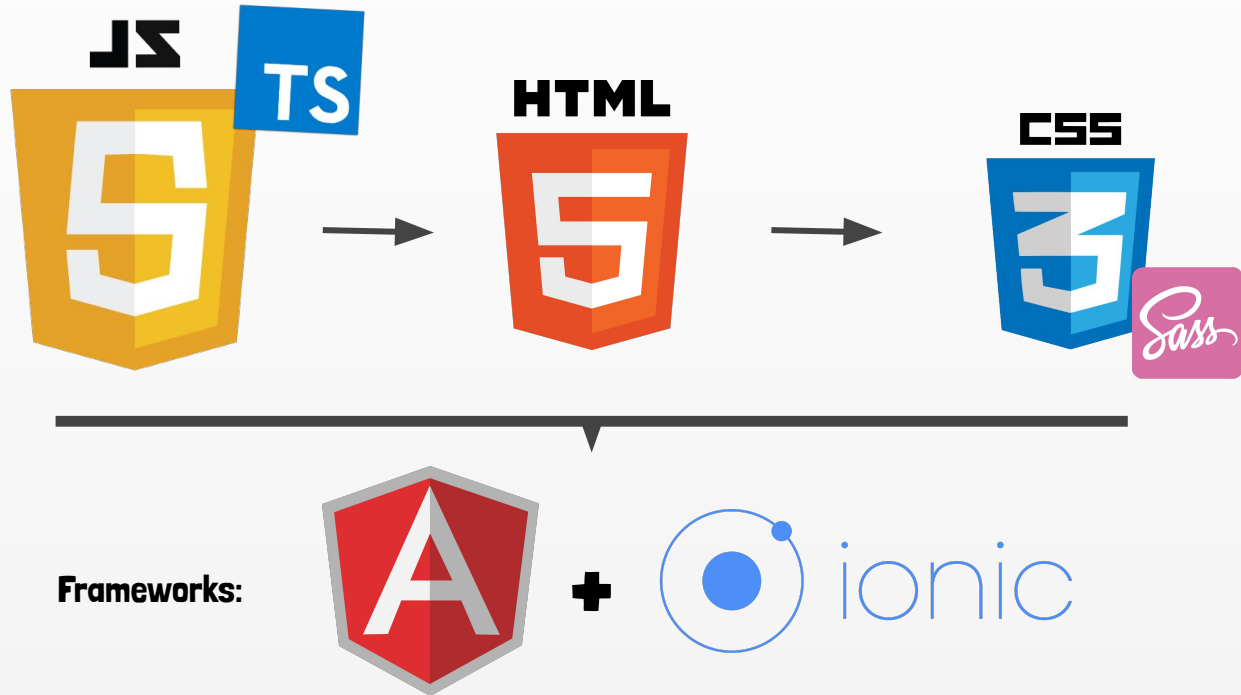
The AWS Cloud spans 99 Availability Zones within 31 geographic regions around the world, with announced plans for 12 more Availability Zones and 4 more AWS Regions in Canada, Israel, New Zealand, and Thailand.





# Languages and frameworks

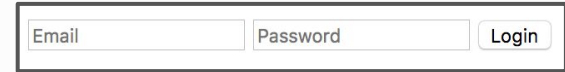
# Business logic, structure and presentation: the languages involved in the **front-end**



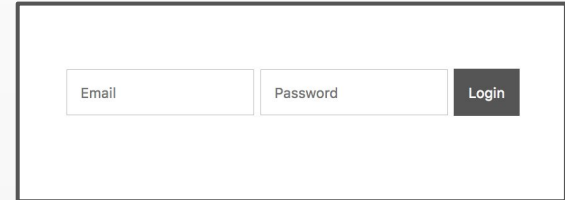
# Structure, presentation and logic: **an example** (1/2)



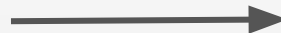
```
<input type="email" name="email" placeholder="Email">
<input type="password" name="pwd" placeholder="Password">
<button onclick="doLogin(email, pwd)">Login</button>
```


```
body {
  text-align: center;
  margin-top: 50px;
}
input {
  padding: 10px;
}
button {
  padding: 12px;
  background-color: #555;
  color: white;
  border: none;
}
```


```
function doLogin(email, pwd) {
  // ...
}
```



***Actual sign-in***

# Structure, presentation and logic: **an example** (2/2)

HTML



```
<ul>
  <li>Lunedì</li>
  <li>Martedì</li>
  <li>Mercoledì</li>
  <li>Giovedì</li>
  <li>Venerdì</li>
  <li>Sabato</li>
  <li>Domenica</li>
</ul>
```



- Lunedì
- Martedì
- Mercoledì
- Giovedì
- Venerdì
- Sabato
- Domenica



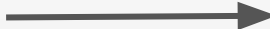
```
let weekDays = ['Lunedì', 'Martedì', 'Mercoledì', 'Giovedì', 'Venerdì', 'Sabato', 'Domenica'];
```

```
<ul>
  <li *ngFor="let day of weekDays">{{ day }}</li>
</ul>
```



```
let weekDays = ['Lunedì', 'Martedì', 'Mercoledì', 'Giovedì', 'Venerdì', 'Sabato', 'Domenica'];
```

```
<ion-list>
  <ion-item *ngFor="let day of weekDays">{{ day }}</ion-item>
</ion-list>
```



Week days
Lunedì
Martedì
Mercoledì

Week days
Lunedì
Martedì
Mercoledì

# Back-end languages & frameworks



... in the back-end too,  
thanks to the framework...

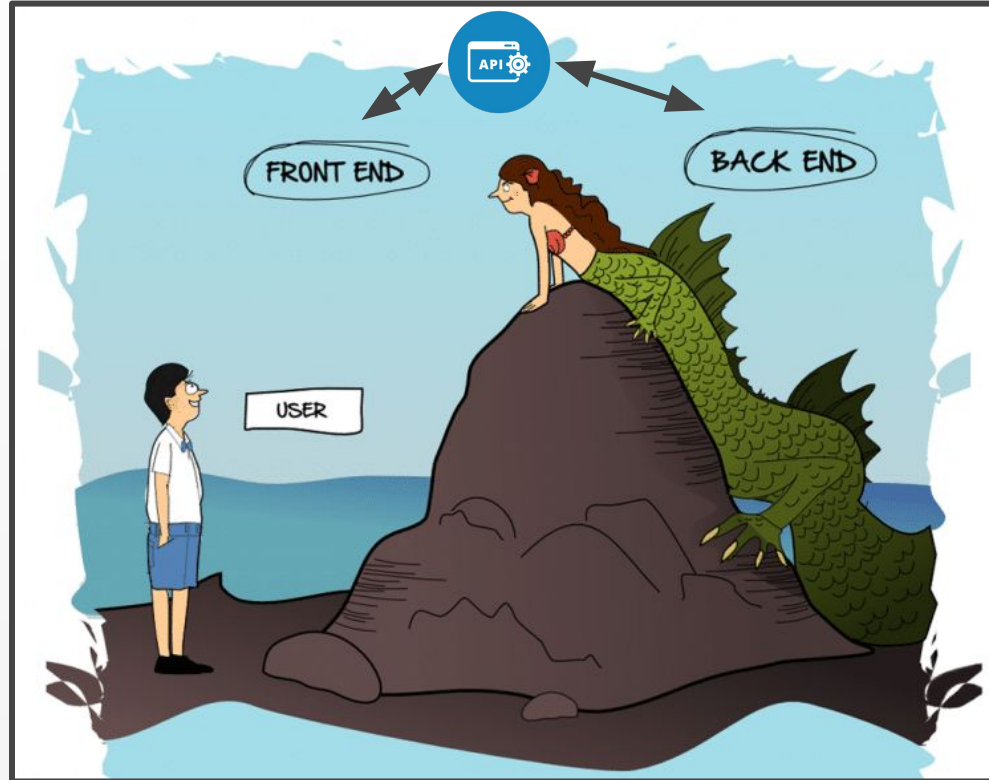


# **Architecture, in practice**

*(... in the next presentation)*

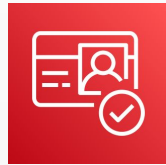
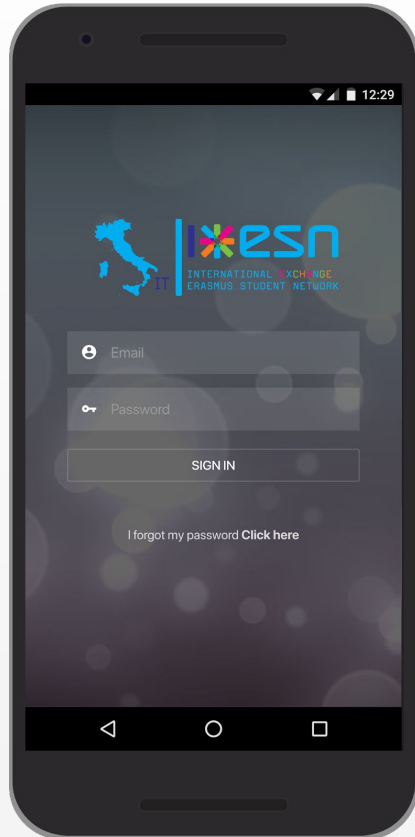
# Architecture, in practice

# Front-end, back-end & API: end-user perspective





# Authentication. Sign-in, forgot password, etc.



**AWS Cognito:**  
*authentication*

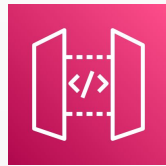
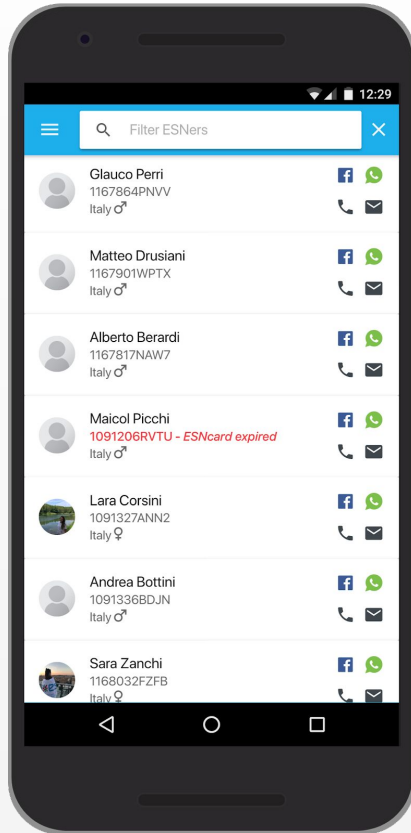
aws User Pools | Federated Identities

### jupiter\_users

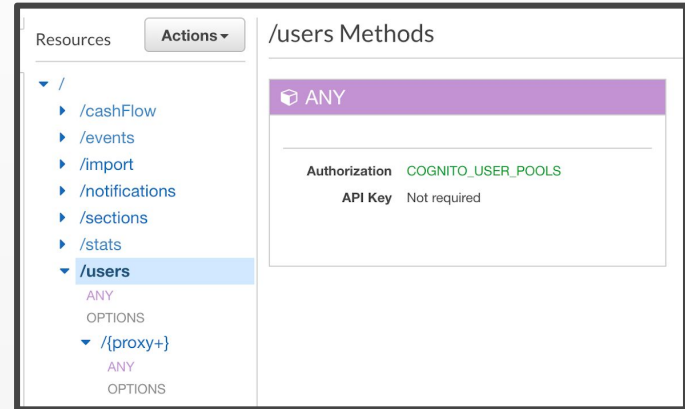
- General settings
  - Users and groups Pool Id eu-we
  - Attributes Pool ARN arn:aws
  - Policies
  - MFA and verifications
  - Advanced security
  - Message customizations
  - Tags
  - Devices
  - App clients
  - Triggers
  - Analytics
- App integration
  - App client settings
  - Domain name

Estimated number of users	7,714
Required attributes	none
Alias attributes	none
Username attributes	email
Custom attributes	Choo
Minimum password length	8
Password policy	no req

# Get users list. API request and data retrieval (1/3)



**AWS API Gateway:**  
API management,  
resources and methods



# Get users list. API request and data retrieval (2/3)

Resources Actions ← Method Execution /users - ANY - In

Provide information about the target backend that th

Integration type  Lambda F

HTTP ⓘ

Mock ⓘ

AWS Ser

VPC Link

Use Lambda Proxy integration  ⓘ

Lambda Region eu-west-2

Lambda Function **jupiter\_users**



**AWS Lambda function:**  
*Request execution*

Lambda > Functions > jupiter\_users

**jupiter\_users**

Configuration | Monitoring

► Designer

Function code [Info](#)

Method	Meaning
GET	Read data
POST	Insert data
PUT or PATCH	Update data, or insert if a new id
DELETE	Delete data

Following **REST logic**, there is a  
Lambda Function for each  
accessible resource

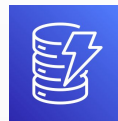
# Get users list. API request and data retrieval (3/3)

```

async getResources(): Promise<User[]> {
  if (!this.requestingUser.permissions.canGetUsers) throw new Error('Unauthorized');

  return await this.dynamoDB.query({
    TableName: this.tables.users,
    KeyConditionExpression: 'sectionId = :sectionId',
    ExpressionAttributeValues: {':sectionId': this.requestingUser.sectionId }
  });
}

```



**AWS DynamoDB:**  
*Database (NoSQL)*



Results sent to the front-end



**AWS CloudWatch:**  
*Execution control*

```

▶ 13:49:27 START RequestId: c11509c9-f4a6-11e8-807e-6d08083268c8 Version: 38
▶ 13:49:27 2018-11-30T13:49:27.600Z c11509c9-f4a6-11e8-807e-6d08083268c8 [START] { resource: '/users', path: '/'
▶ 13:49:27 2018-11-30T13:49:27.623Z c11509c9-f4a6-11e8-807e-6d08083268c8 ..... GET jupiter_users » { Item: { do
▶ 13:49:27 2018-11-30T13:49:27.626Z c11509c9-f4a6-11e8-807e-6d08083268c8 [REQUEST] GET
▶ 13:49:28 2018-11-30T13:49:28.154Z c11509c9-f4a6-11e8-807e-6d08083268c8 ..... QUERY jupiter_users » 257
▶ 13:49:28 2018-11-30T13:49:28.154Z c11509c9-f4a6-11e8-807e-6d08083268c8 [DONE] [ { domicile: { city: null, cour
▶ 13:49:28 END RequestId: c11509c9-f4a6-11e8-807e-6d08083268c8

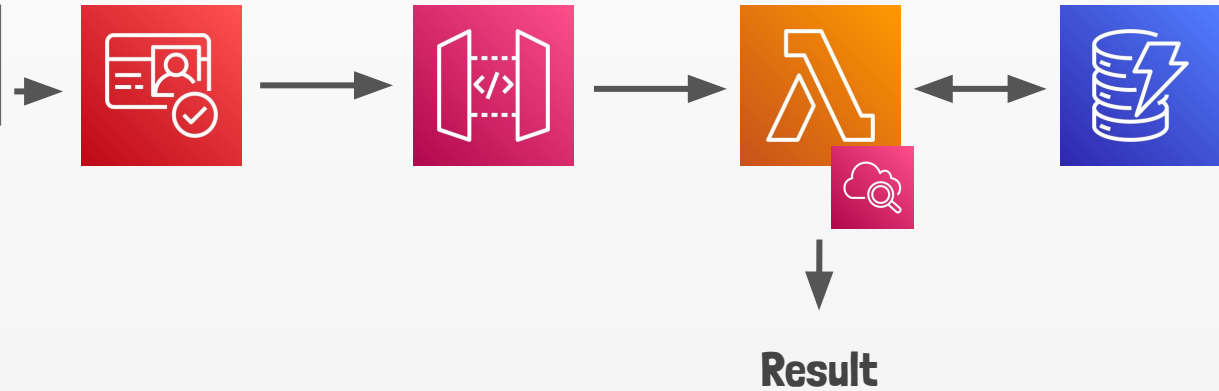
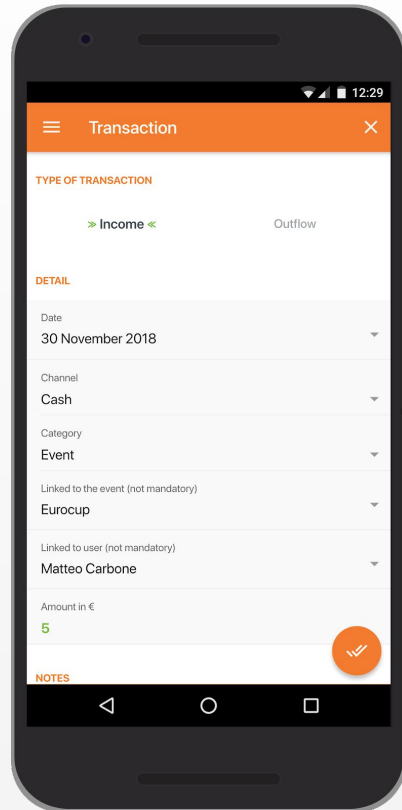
```

Name	userid
idea_JUID	
jupiter_sections	
jupiter_sections_cashFlowStatement	
jupiter_sections_events	
jupiter_sections_eventsArchived	
jupiter_sections_usersArchived	
jupiter_sectionsEvents_users	
jupiter_stats	
<b>jupiter_users</b>	

userid
0110cb5b-4f13-4b15-96a6-a5d448265f7b
02d5e0ce-e2a0-45a8-b39f-53e54a2f9457
03722561-7bb4-4794-8111-3a470da818e8

# Post new financial transaction. Another example



## Other useful services



**AWS S3:**  
*File storage*



**AWS IAM:**  
*Permissions to access  
resources*



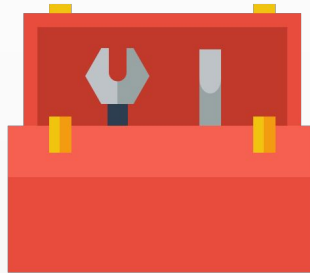
**AWS SES:**  
*Send emails*

*...and much more!*

## IDEA's extra modules



**idea-aws**



**idea-toolbox**

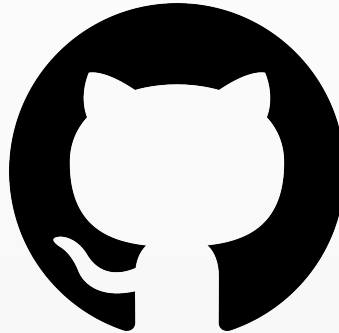


**IDEA-Ionic-Extra**

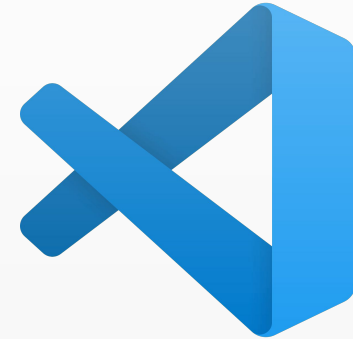
# Our tools



**Slack**  
Communication



**GitHub**  
Code repository



**Visual Studio Code**  
IDE



# From web to **mobile stores**

# From web to **mobile stores**



# tldr;

The **recipe**  
for our cloud solutions

- Process: **IDEA** → **product**
- **Front-end, back-end** (& API)
- **Cloud architecture**  
(**serverless**) on Amazon Web Services - *next presentation*
- **Languages and frameworks**
- **App** on mobile stores

[ITER-IDEA.com/blog](https://iter-idea.com/blog)

