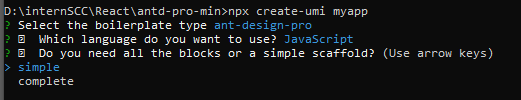
**BÁO CÁO 12: TÌM HIỂU ANT DESIGN PRO**

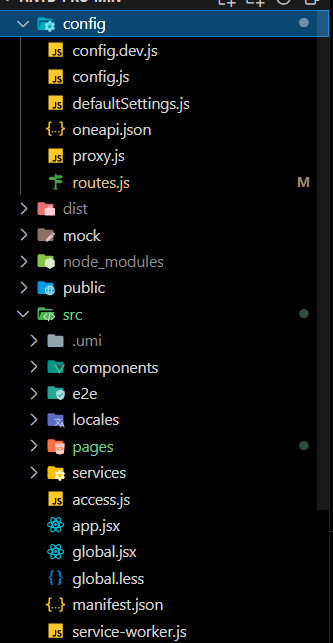
1. **Ant design pro**
2. **Cài đặt**

* Cài đặt Ant design pro khá đơn giản với lệnh npx create-umi myapp và em chọn cài đặt bản single sử dụng ngôn ngữ JavaScript

****

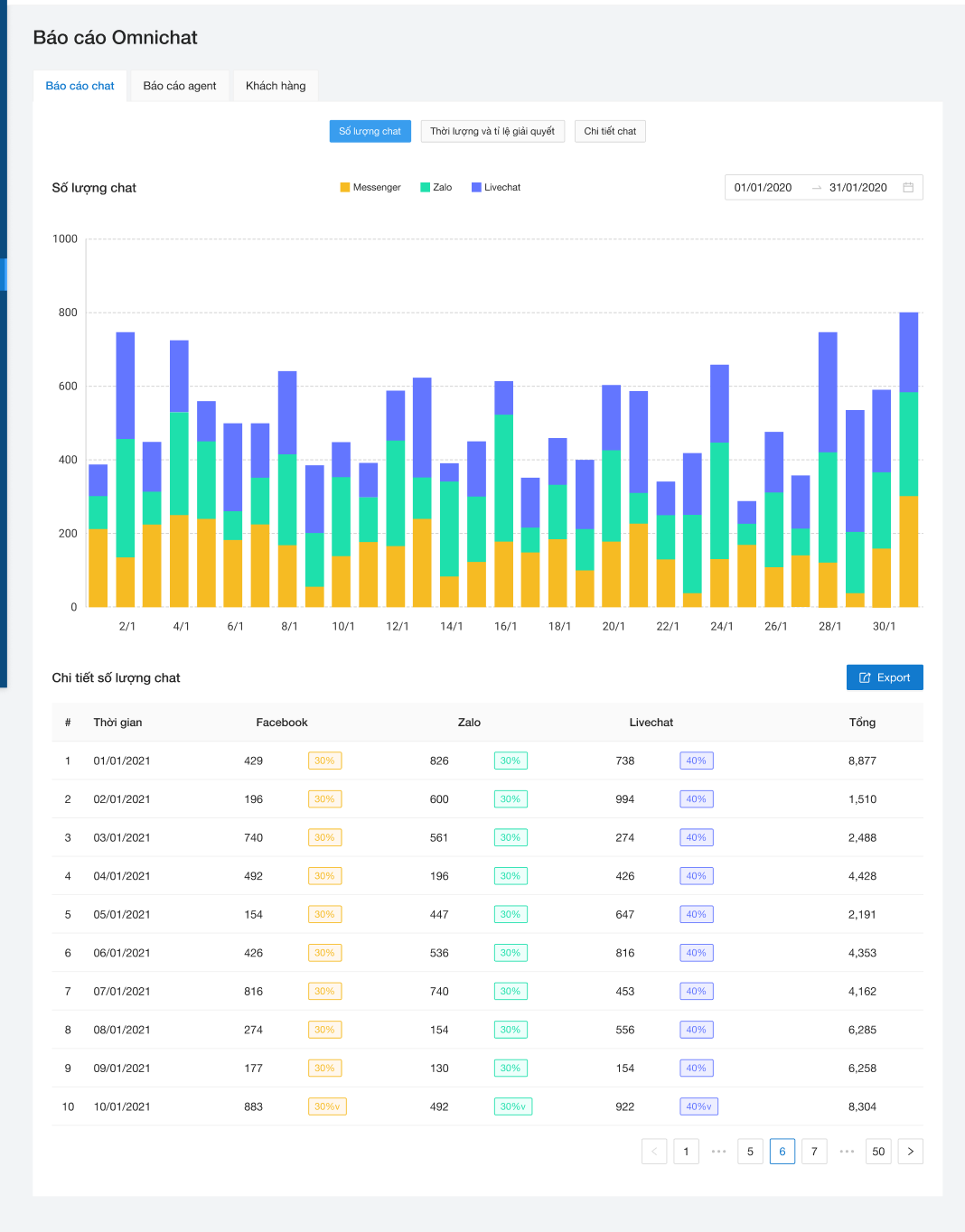
1. **Cấu trúc thư mục**

* Cấu trúc thư mục của project sau khi cài đặt như sau:

****

* Ant design pro sử dụng UmiJS và DvaJS. Vì thế, việc routing sẽ được cài đặt trong file ***routes.js*** và config những thứ khác ở trong file ***config.js***.
* Ở đây em sẽ quan tâm đến 3 folder là *components,* *pages* và *services  :*
  + Folder *components* sẽ chứa các components sử dụng chung
  + Folder *pages* sẽ chứa các trang trong project
  + Folder *services* sẽ chứa các thủ tục gọi api

1. **Demo bizchart và tạo table chi tiết số lượng chat trên Ant design pro**

* Em sẽ thử tạo 1 trang sử dụng bizchart và table như sau:

1. **Cài đặt chung**

* Đầu tiên em sẽ tạo ra một trang báo cáo omni có các thẻ như *Báo cáo chat, Báo cáo agent, Khách hàng* để chứa chart và table
* Trong file ***/reports/index.js*** :

import { Tabs } from 'antd';

import ReportChat from './reportChat';

import './index.less';

const { TabPane } = Tabs;

const reportOmni = () => {

  function callback(key) {

    console.log(key);

  }

  return (

    <div>

      <h3>Báo cáo Omnichat</h3>

      <div className="card-container">

        <Tabs onChange={callback} type="card">

          <TabPane tab="Báo cáo chat" key="1">

            <ReportChat />

          </TabPane>

          <TabPane tab="Báo cáo agent" key="2">

            Content of Tab Pane 2

          </TabPane>

          <TabPane tab="Khách hàng" key="3">

            Content of Tab Pane 3

          </TabPane>

        </Tabs>

      </div>

    </div>

  );

};

export default reportOmni;

* Sau đó em sẽ thêm các thẻ *Số lượng chat, Tỉ lệ và thời lượng giải quyết* và C*hi tiết chat* trong thẻ *Báo cáo chat*
* Trong file ***/reports/reportChat/index.js***:

import { Tabs } from 'antd';

import CountChat from './countChat';

const { TabPane } = Tabs;

const ReportChat = () => {

  function callback(key) {

    console.log(key);

  }

  return (

    <div>

      <Tabs defaultActiveKey="1" type="card" centered onChange={callback}>

        <TabPane tab="Số lượng chat" key="1">

          <CountChat />

        </TabPane>

        <TabPane tab="Thời lượng và tỉ lệ giải quyết" key="2">

          Content of Tab Pane 2

        </TabPane>

        <TabPane tab="Chi tiết chat" key="3">

          Content of Tab Pane 3

        </TabPane>

      </Tabs>

    </div>

  );

};

export default ReportChat;

* Sau đó, em sẽ tạo một component để chứa bizchart và table chi tiết số lượng chat
* Trong file ***/reports/reportChat/countChat/index.js***:

import Chart from './components/chart';

import Table from './components/table';

const CountChat = () => {

  return (

    <div>

      <Chart />

      <Table />

    </div>

  );

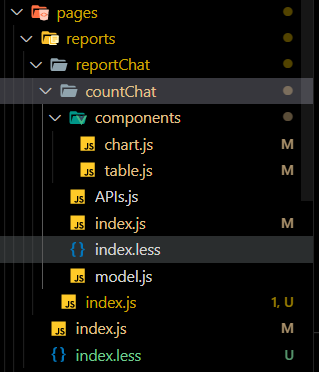
};

export default CountChat;

* Bây giờ em sẽ tạo chart và table

1. **Demo bizchart**

* Trước tiên, em sẽ chia folder của mình như sau

****

* Vì ở đây em không có API cụ thể để lấy data nên em sẽ tạo data giả định với cấu trúc như sau :

const data = [

  {

    id: 1,

    date: '18/04/2021',

    type: 'Zalo',

    amount: 100,

  },

  {

    id: 2,

    date: '19/04/2021',

    type: 'Zalo',

    amount: 200,

  },

  {

    id: 3,

    date: '20/04/2021',

    type: 'Zalo',

    amount: 300,

  }, …

* Trong file ***chart.js*** em sẽ tạo chart sử dụng bizchart 4.0:

registerInteraction('element-link', {

  start: [{ trigger: 'interval:mouseenter', action: 'element-link-by-color:link' }],

  end: [{ trigger: 'interval:mouseleave', action: 'element-link-by-color:unlink' }],

});

const ChartComponent = () => {

  return (

    <div>

      <Space direction="vertical" size={12}>

        <RangePicker />

      </Space>

      <Chart padding="auto" autoFit height={500} data={data}>

        <Tooltip shared showMarkers={false} />

        <Interval position="date\*amount" color="type" adjust="stack" />

        <Legend position="top" />

        <Interaction type="element-highlight" />

        <Interaction type="element-link" />

      </Chart>

    </div>

  );

};

export default ChartComponent;

1. **Table**

* Tương tự với chart em cũng sẽ tạo data giả định như sau :

const data = [

  {

    id: 1,

    date: '01/01/2021',

    mess: 315,

    zalo: 123,

    livechat: 645,

    sum: 1083,

  },

  {

    id: 2,

    date: '02/01/2021',

    mess: 334,

    zalo: 625,

    livechat: 244,

    sum: 1203,

  },

];

* Trong file ***Table.js***em sẽ quy định table có các cột như sau:

const columns = [

  {

    title: '#',

    dataIndex: 'id ',

    key: 'id',

  },

  {

    title: 'Thời gian',

    dataIndex: 'date',

    key: 'date',

  },

  {

    title: 'Facebook',

    dataIndex: 'mess',

    key: 'mess',

    render: (text, record) => (

      <div>

        <Row>

          <Col span="12"> {text}</Col>

          <Col span="12">

            <Tag color="volcano">{Math.round((text / record.sum) \* 100)}%</Tag>

          </Col>

        </Row>

      </div>

    ),

  },

  {

    title: 'Zalo',

    key: 'zalo',

    dataIndex: 'zalo',

    render: (text, record) => (

      <div>

        <Row>

          <Col span="12"> {text}</Col>

          <Col span="12">

            <Tag color="green">{Math.round((text / record.sum) \* 100)}%</Tag>

          </Col>

        </Row>

      </div>

    ),

  },

  {

    title: 'Livechat',

    key: 'livechat',

    dataIndex: 'livechat',

    render: (text, record) => (

      <div>

        <Row>

          <Col span="12"> {text}</Col>

          <Col span="12">

            <Tag color="geekblue">{Math.round((text / record.sum) \* 100)}%</Tag>

          </Col>

        </Row>

      </div>

    ),

  },

  {

    title: 'Tổng',

    key: 'sum',

    dataIndex: 'sum',

  },

];

const TableComponent = () => {

  return (

    <div>

      <div>

        <Row>

          <Col flex="300px">

            <b>Chi tiết số lượng chat</b>

          </Col>

          <Col flex="1100px"></Col>

          <Col flex="auto">

         <Button type="primary" icon={<ExportOutlined />}>

           Export

         </Button>

          </Col>

        </Row>

      </div>

      <Table columns={columns} dataSource={data} />

    </div>

  );

};

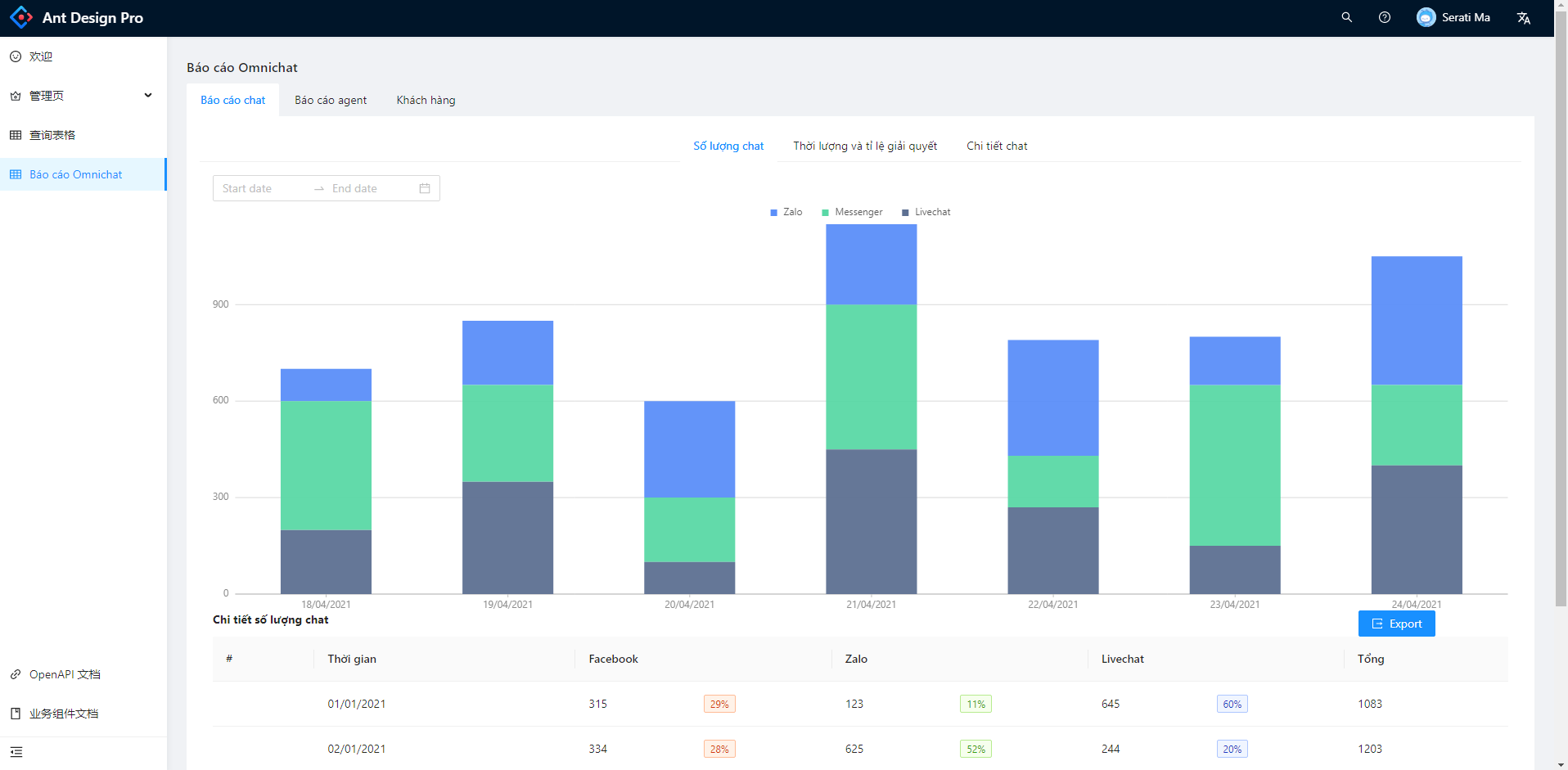
export default TableComponent;

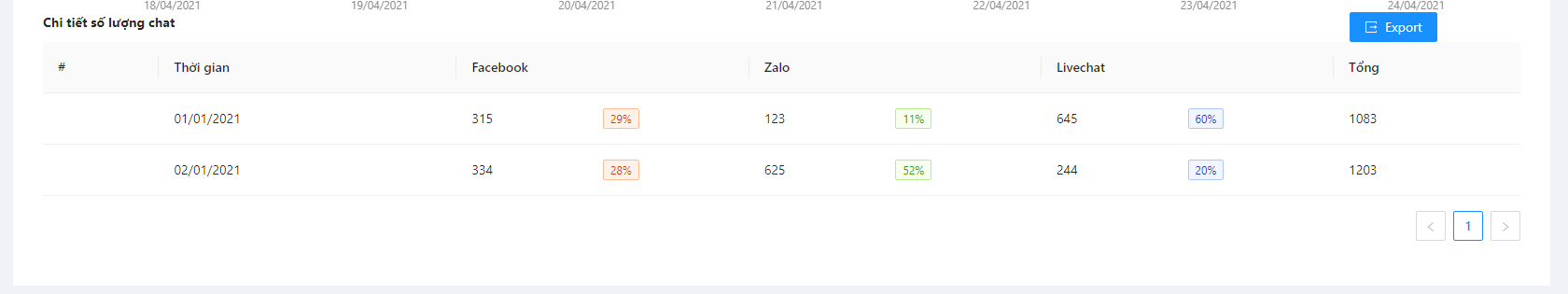
1. **Source**

* Tất cả source code nằm trong link github sau:

[internSCC/React/antd-pro-min at main · uytran36/internSCC (github.com)](https://github.com/uytran36/internSCC/tree/main/React/antd-pro-min)

1. **Demo**

****

****