
Read the Docs Template Documentation

Выпуск 0.1-alpha

Read the Docs

февр. 06, 2023

Оглавление

Для правильной работы приложения SynergyApp и авторизации в подключенных к нему приложений конструктора необходимо:

1. В приложении SynergyApp открыть скрипт `synergy_app_page.js`

В скрипте есть массив приложений `const synergyApps = [];`

Необходимо добавить в данный массив список приложений, например:

```
synergyApps.push({
  name: 'Потоки работ',
  icon: 'work', // https://fonts.google.com/icons?icon.style=Filled&icon.set=Material+Icons
  url: 'workflows' // url приложения в конструкторе
});
```

Рис. 1: Рис. Добавление приложения в список

2. Иконку приложения необходимо брать с `fonts.google.com`

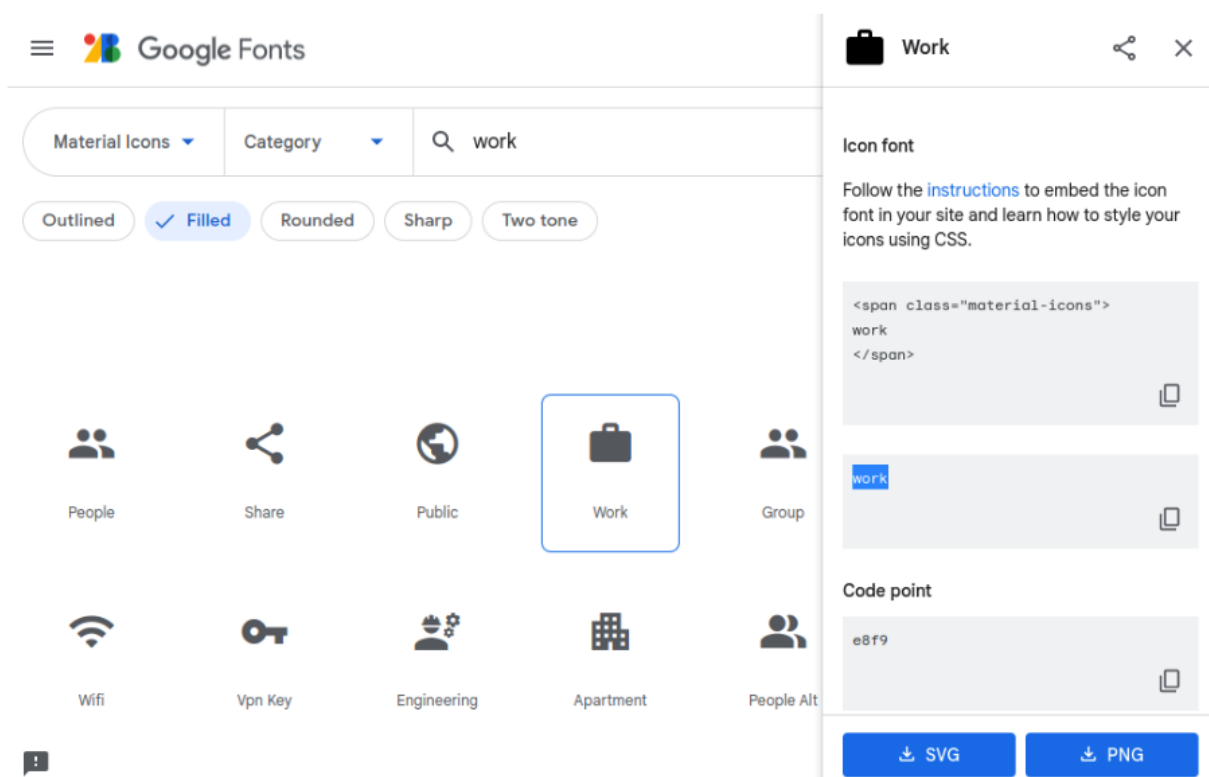


Рис. 2: Рис. Добавление иконки

3. Настройка приложения для подключения в качестве модуля к SynergyApp

Необходимо будет создать пустую страницу “заглушку”, сделать ее стартовой и страницей авторизации:

Код страницы указать, например, `start_page`

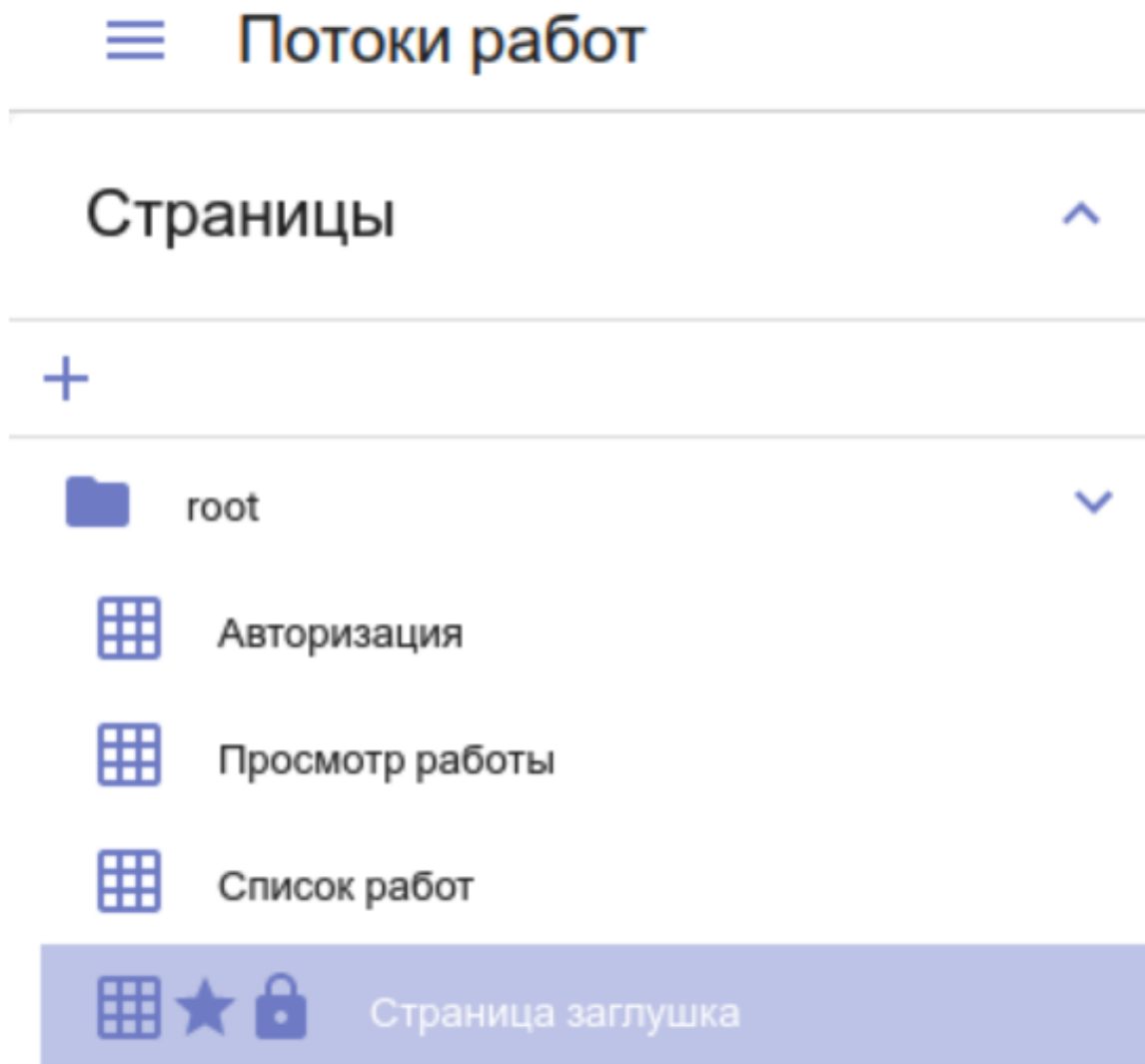


Рис. 3: Рис. Добавление заглушки

После чего необходимо создать скрипт для этой страницы, который будет обрабатывать переданные данные для авторизации из приложения SynergyApp.

Внимание! Если есть кейсы на успешную авторизацию, их необходимо убрать.

```
const getUrlParameter = sParam => {
  let sPageURL = window.location.search.substring(1), sURLVariables = sPageURL.split('&'),
  sParameterName;
  for (let i = 0; i < sURLVariables.length; i++) {
    sParameterName = sURLVariables[i].split('=');
    if (sParameterName[0] === sParam)
      return sParameterName[1] === undefined ? true : decodeURIComponent(sParameterName[1]);
  }
}

pageHandler('start_page', () => {
  const app = getUrlParameter('app');
  const locale = getUrlParameter('locale');

  if(locale) {
    localStorage.locale = locale;
    fire({type: 'change_locale', locale});
  }

  if(app) {
    Cons.setAppStore({appInIframe: true});
    const basicParams = btoa(encodeURIComponent(decodeURIComponent(unescape(atob(app)))));
    const decodeStr = decodeURIComponent(atob(basicParams));
    const separator = decodeStr.slice(0,8);
    const authParams = decodeStr.split(separator);
    Cons.login({
      login: authParams[1].split('').reverse().join(''),
      password: authParams[2].split('').reverse().join('')
    });
  } else {
    fire({type: 'goto_page', pageCode: 'authPage'}, 'root-panel');
  }

  if (!Cons.getAppStore().start_page_auth_listener) {
    addListener('auth_success', 'root-panel', authed => {
      const {login, password} = authed.creds;

      Cons.creds.login = login;
      Cons.creds.password = password;
      AS.apiAuth.setCredentials(login, password);
      AS.OPTIONS.login = login;
      AS.OPTIONS.password = password;
      AS.OPTIONS.currentUser = authed.data.person;
      fire({type: 'goto_page', pageCode: 'listWorksPage'}, 'root-panel');
    });

    Cons.setAppStore({start_page_auth_listener: true});
  }
});
```

4. Прописать в свойствах приложений SynergyApp и связанных с ним приложениях логин и пароль служебного пользователя с правами администратора.
5. Остается задеплоить своё приложение и приложение SynergyApp.

p.s. дизайн страницы авторизации и верхней панели приложения SynergyApp остается за вами