**Документация**

**Използвани технологии**

* **BackEnd (server-side)**
* Node.js
* Express.js - уеб фреймуорк за лесна разбработка на Node.js сървърни приложение.
* MongoDb - NoSQL документ базирана база от данни за съхранение на обработените от сървъра данни.
* Mongoose - Node.js библиотелка за работа с MongoDb която предлага модели и валидации на отделните документи.
* **FrontEnd (client-side)**
* React.js
* Axios - библиотека за осъществяване на REST комуникацията посредвтвом клиент-сървър.
* React Router - маршрутизация на отделните изгледи
* Formik - работа с форми и лесна обработка на въведените данни, чрез интеграция с yup се прилагат и валидации.
* Yup - за валидация на отделните полета

**REST API маршрути**

* GET /recipes - списък рецепти
* POST /recipes/create - създаване на рецепта
* GET /recipes/:id - рецепта
* PUT /recipes/:id - обновяване на рецепта
* DELETE /recipes/:id - изтриване на рецепта
* GET /comments/:recipeId - списък коментари за конкретна рецепта
* POST /comments/:recipeId - добавяне на коментар за рецепта
* DELETE /comments/:recipeId - изтирване на коментар за рецепта
* POST /auth/signin - създаване на потребител
* POST /auth/signup - Вход
* GET /users - списък със всички активни потребители
* GET /users/:id - потебител
* PUT /users/:id - обноваване на потребител
* GET /users/:id/recipes - рецепти от даден потребител
* GET /users/:id/comments - коментари от даден потребител

**FrontEnd изгледи**

* Recipes - списък рецепти сортирани по дата на създаване
* Recipes/:Recipe - рецепта
* Search - търсене на рецепта
* Users - списпък на активни потребители
* Users/:User - публикувани рецепти от потребител
* Recipes/Create - създаване на рецепта
* Recipes/Update - обновяване на рецепта
* Dashboard - управление на акаунта