**3. Proposed Software Architecture**

Survey4All is a system which you can create surveys, distribute them to public view and analyze the results. We wanted to make sure that people can create their own survey and answer the others’ surveys easily, that’s why we designed a user friendly, simple interface. The Survey4All System has 3 actors: Registered User, Unregistered User, Admin. Users can register to the system all free. Without registration, you can view and share surveys but you can not fill a survey, create/delete/edit your survey or favorite surveys. You can edit your profile and see your surveys on your profile page if you are a registered user. Survey and user informations are stored in the database. The data is retrieved from our database provider Firebase by queries.

**3.1. Overview**

The system will have a layered architecture and there will be 3 layers, which are:

1. Interface
2. Application
3. Storage

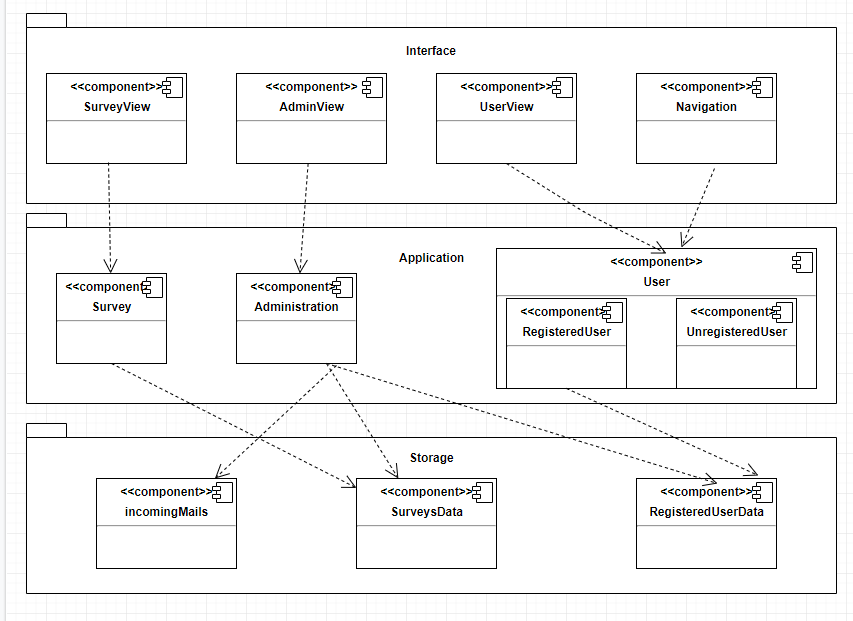
Users will be interacting with the system using the interface layer, application layer will have all the functions of the system in it and storage layer will have all the data about the users and surveys. Storage layer will have the registeredUserData, SurveysData and incomingMails. Subsystems retrieve these information from our database. Application layer will have the survey, admin, user(unregistered and registered user) subsystems. User subsystem will have the functions of register, login, searching a on the system, editing the user profile and sending messages to the website admin. User subsystem keeps some of the functions for both unregistered and registered user such as viewing surveys but some of the functions are only for registered users such as favorite survey, delete survey, create survey, edit profile.

* Registered user subsystem will keep the functions of login, editing profile, messaging, create a survey, fill a survey, favorite a survey.
* Unregistered user subsytem has access to view surveys, fill a survey, search for a survey, share survey.
* Survey subsystem handles the surveys created by registered user, will have the functions of giving a respond to the survey.
* Admin subsystem will have the functions of managing a survey such as approving or

rejecting the auction, managing the user profile such as banning their

account, delete inappropriate surveys. Admin has and managing the contact us messages.

* 1. **System Decomposition**

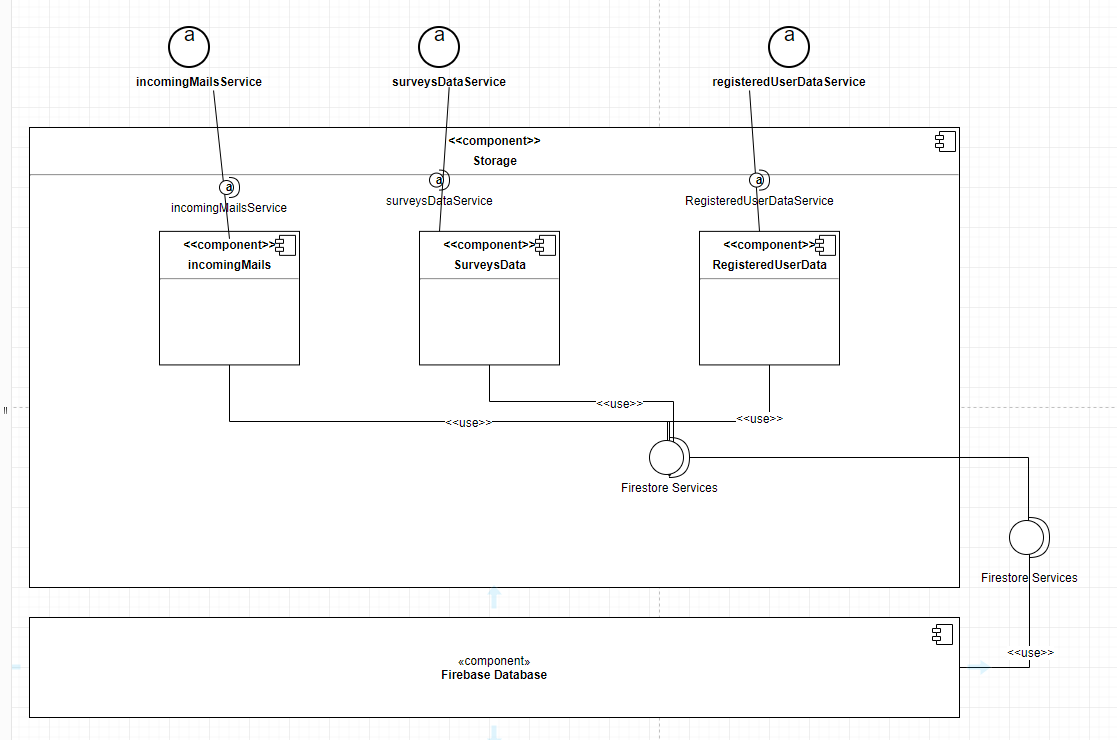


Storage layer holds the RegisteredUserData, SurveysData and incomingMails subsystems. RegisteredUserData has the information (e-mail, name, phone number, user type…) about the Survey4All users which we define as RegisteredUsers and the Admin.

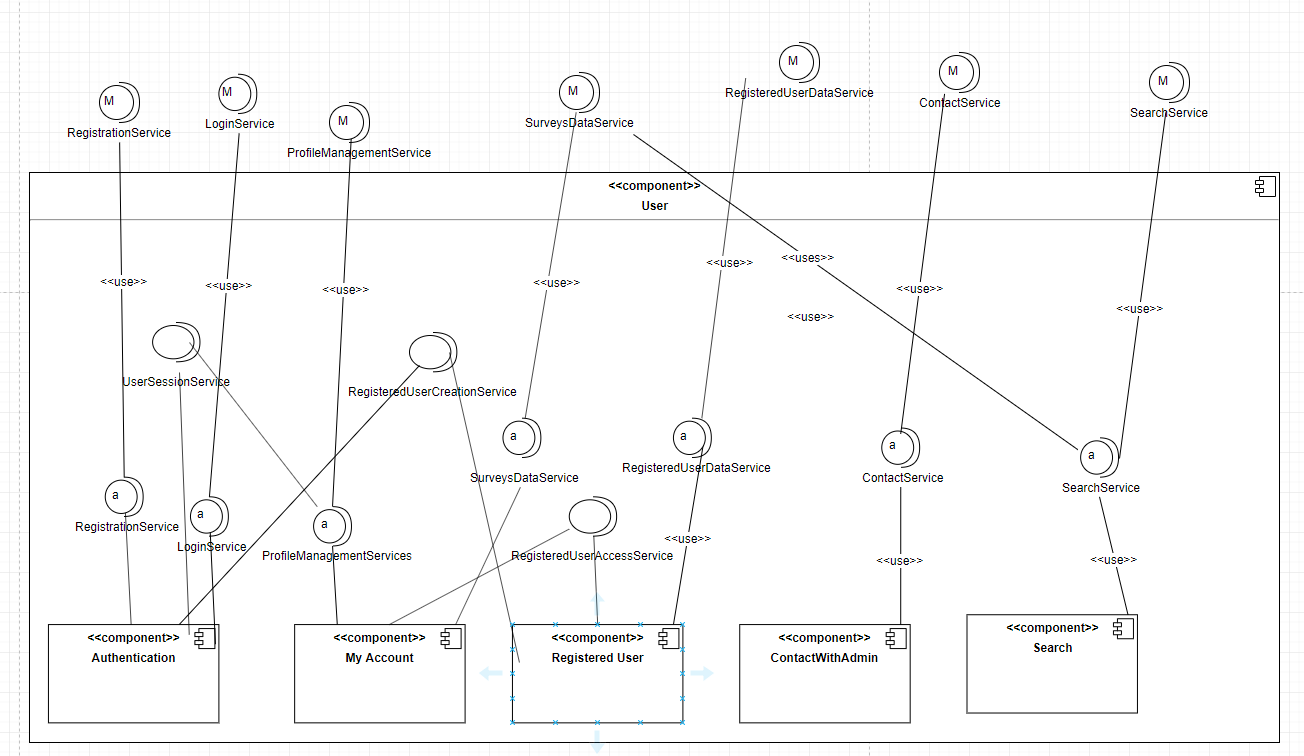
Application layer holds Survey, Administration and User subsystems. User subsystem has 2 different subsystems in it: RegisteredUser, UnregisteredUser. User subsystem is the subsystem that manages all the common services that a user needs. It is about user functionalities like messaging, profile management, search surveys and log-in/log-out services. Administration subsystem is the subsystem that manages the all admin privilages like deleting surveys, viewing contact us page mails, or even banning a user’s account due to his/her usage of the system. Survey subsystem has all the functionalities any survey can proceed such as creation of a survey, deletion of a survey, extraction and analysis of the results of a survey.

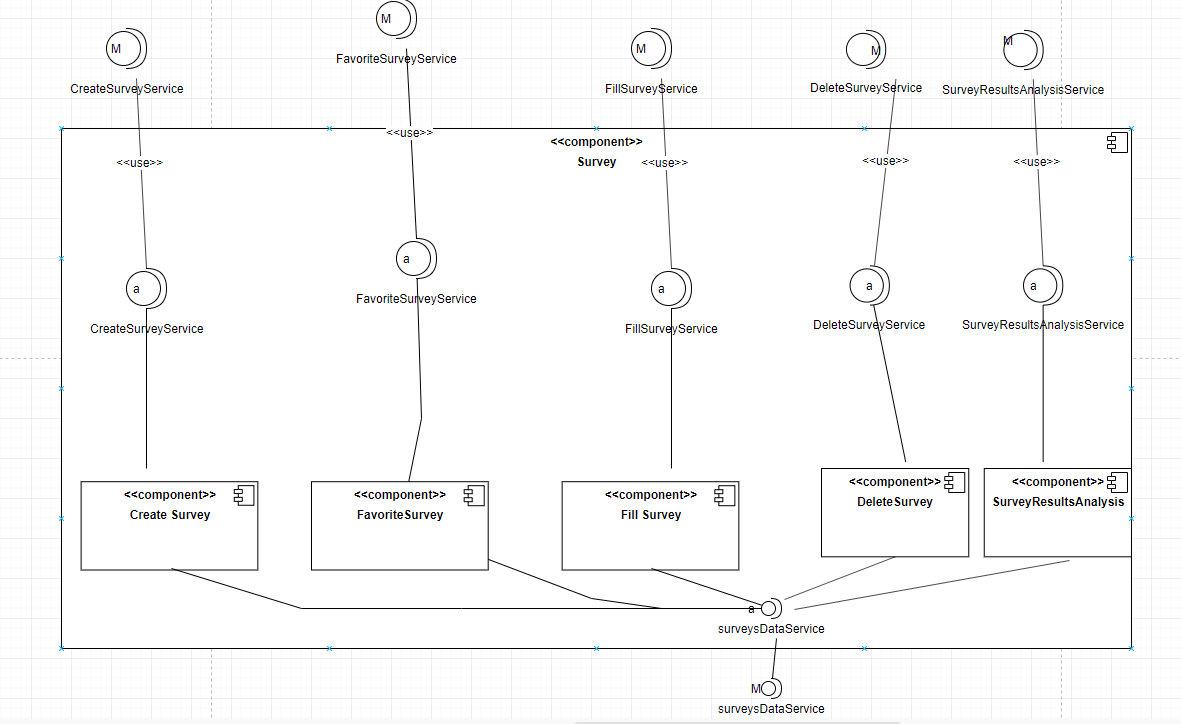
As the last of our layers, Interface layer has the boundary objects, all user interface that users interact with. It holds SurveyView, AdminView, Navigation and UserView. SurveyView is the subsystem that controls the ongoing or finished survey services. AdminView is essential for admin to manage common services that stabilizes and holds the control users and surveys for the sake of the system. Navigation is the primary for services like login search and register. It has routing service directly or indirectly (For ex: by pressing a button or navbar tab) UserView has services that giving users to manage simple actions. Users with the userType: user, unregistered users and the admin see different pages because of the authentication service which Firebase provides us.

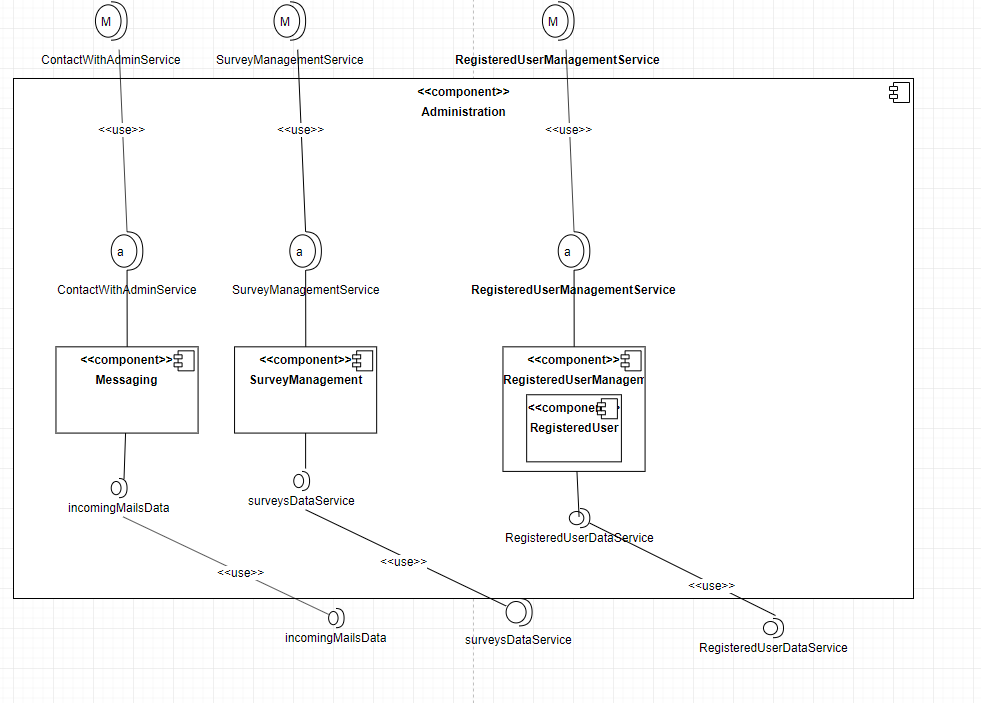
We use Firebase’s Firestore Database which is a NOSQL database. So in order to get a data we type the query Firestore Service provides us. In the Storage subsystem we collect all necessary data such as incomingMails, RegisteredUserData, surveysData and connect to the Database system via Firestore Service.



The User subsystem provides registeredUsers and their functions such as registration, login , message admin, edit profile, create survey, favourite survey.



The Survey subsystem has following functions: Create survey, delete survey, fill survey, favorite survey, survey results analysis service”



The Administration Subsystem handles Admin’s functionalities, such as management of registeredUsers (banning their accounts, deleting inappropriate surveys etc.), survey information, viewing received messages from RegisteredUsers or UnRegisteredUsers on contact us form.

Here is the interface layer diagrams containing:

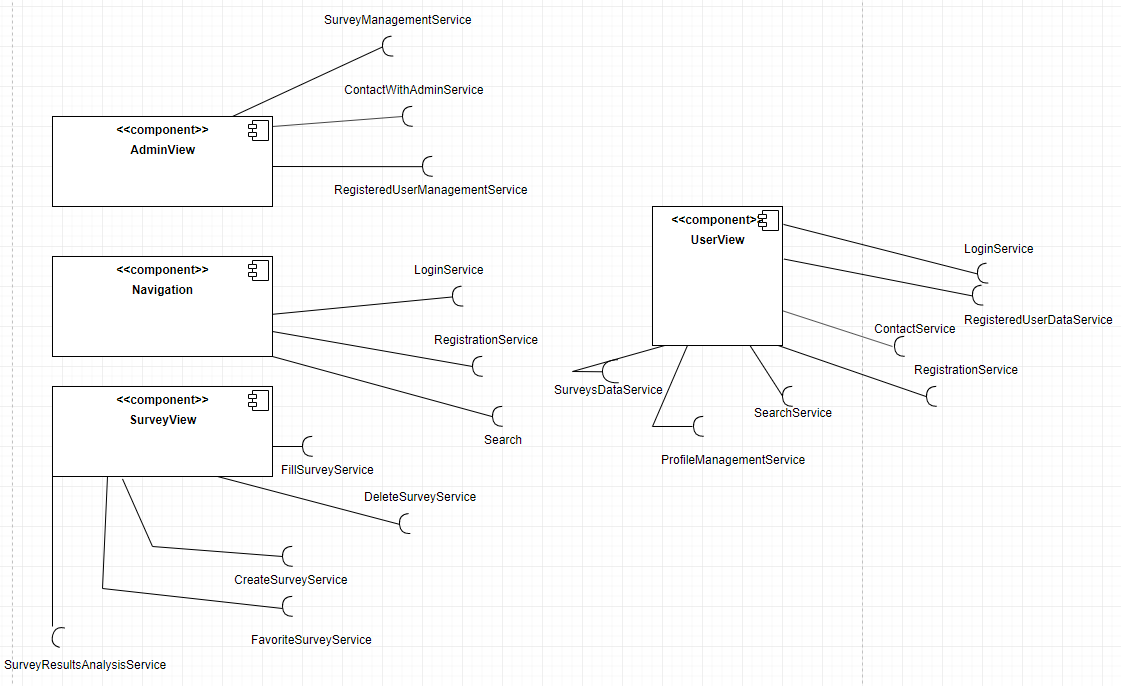
-User View

-AdminView

-SurveyView

-Navigation

User type selects which interface will be shown.



Our system is a web application that is written with Angular, and interface layer holds the view users can interact with. It holds 3 View subsystems for each application layer subsystem. It also holds the Navigation subsystem which holds the pages like homepage, register and login, browse surveys, contact us.