1st Sprint – Web Application "Sami Rooms"

Benyamin Yakobi 323492835  
Sagi Vaknin 316605617  
Liron Vaknin 204447882   
Kesem Even-Hen 208055483  
Dan Marinescu 204264543

1. [Project Documentation](#project_docs)
   1. [Functional Requirements](#fun_req)
      1. [[Website Login Epic](#fun_req)](#Website_Login_Epic)
      2. [[Student Homepage Epic](#fun_req)](#Student_Homepage_Epic)
      3. [[Renter Homepage Epic](#fun_req)](#Renter_Homepage_Epic)
      4. [[Booking Management Epic](#fun_req)](#Booking_epic)
      5. [[Orders Management Epic](#fun_req)](#Orders_epic)
      6. [[Payment Window Epic](#fun_req)](#paymentEpic)
      7. [[Website Support Epic](#fun_req)](#Website_support_epic)
      8. [Backend Management Epic](#Backend_Management_Epic)
   2. [Non-Functional Requirements](#non_func_reqs)
      1. [Security](#security)
      2. [Performance](#perofrmance)
      3. [Scalability](#scalability)
      4. [Efficiency](#efficiency)
      5. [Durability](#durability)
      6. [Availability](#availability)
   3. [Test-Cases](#testCases)
2. [Project Infrastructure](#Project_infrastructure)
   1. [Conventions](#convention)
   2. [GitHub](#github)
   3. [ClubHouse](#clubhouse)
   4. [Slack](#slack)
   5. [CircleCi](#circleci)
   6. [Firebase](#firebase)
3. **Project Documentation**
   1. Functional Requirements
      1. **Website Login Epic** – User shall be able to Log-In as a Student or as a Renter.  
           
         User-Stories:
         1. As a user, I could login to the website in order to rent\post an apartment.
         2. As a user, I can edit my personal information, so that the system will be up to date with my information.
         3. As a user, I would like to have option to recover my account password/username in case I have forgot one of the details.
         4. As a user, I can open a new renter account, which can post and rent apartments for students.
         5. As a user, I can open a student account which allows me to rent apartment during the semester period.
      2. **Student Homepage Epic** – The Student user shall have a Web Page with custom options that specifically customized for Students.  
           
         User-Stories:
         1. As a user student, I want to be able to log-out after I'm logged in.
         2. As a Student, I want to be able to mark a living-unit as an 'Favorite' and my favorites will be displayed in the right side of the screen to watch them later on.
         3. As a Student user / visitor of the website, I want to be able to sort living-units by number of rooms.
         4. As a Student user / visitor I want to be able to sort the prices of the living-units that are available from higher to lower and vice versa.
         5. As a Student user / visitor of the website, I want to be able to search living-units by their name.
         6. As a Student, I want to see news updates on the front page, so I can keep up with the dorm news.
         7. As a Student, I can view any contract ,in order to avoid misunderstandings.
         8. As a Student , I can view all available units that will display each unit information, so that i can easily choose a living unit that matches my conditions.
         9. As a student, when I update my personal information, it should be displayed immediately in my student profile, to stay up to date .
         10. As a Student, I want to view posts as a table or a gallery, so that i can view all living units easily.
      3. **Renter Homepage Epic** – The Renter user shall have a Web Page with custom options that specifically customized for Renters.  
           
         User-Stories:
         1. As a renter, I can tag the living units for its unique properties, to make it more appealing for the students.
         2. As a renter, I can review all posted apartments, for easy management and reviewing properties.
         3. As a renter, I want to be able to log-out after I am logged in.
         4. As a renter, when I update my personal information, it should be displayed immediately in my living-units posts, to stay up to date.
         5. As a Student, I can see the renter rating and reviews, to get impression about the renter.
         6. As a renter, I can edit information of a living unit that is posted on the website such as price, renting period, etc., so that i could match current real estate market.
         7. As a renter, I could post a living unit that i wish to rent which i could enter the min and max renting period, in order that students will rent the unit.
         8. As a renter, I want to create or edit a contract, so I can avoid misunderstanding.
         9. As a renter, I want option to sort my units by status, so I can view which units has been rented.
         10. As a renter, I want to be able to offer discounts by percentages, to attract new customers.
      4. **Booking Management Epic** – Living-Units that are available for rent shall be presented in 'Booking' homepage.

User-Stories:

* + - 1. As a user / visitor, I want to be able to see automatic recommendations of outstanding living units based on student rating & their comments.
      2. As a user / visitor, I want to be able to see the real price and current price after discount to be able take decision.
      3. As a student, I want to be able to leave a comment with my impressions to give feedback to the renter.
      4. As a student, I want to be able to rate living-units, to help other students.
      5. As a student, I do not want to see apartments which already rented.
      6. As a renter, the availability of the unit must be updated after I have rented it, to avoid misunderstanding with other customers.
    1. **Orders Management Epic** – Each user, either Student or Renter, shall be able to view relevant information which relevant to contracts, contact information or information that relevant to a specific order.

User-Stories:

* + - 1. as a student, I want to be able to view my Renter information to be able to contact him at any time.
      2. As a renter, I want to be able to view my customers (=students) information to be able to contact with them at any time.
      3. As a renter I would like to see the status of all my rental units, in order to review my revenue / expenses.
      4. As a renter, I should be updated when one of my units sold, so I can make my arrangements for the unit.
      5. As a student, I should be able to know when the next due payment, so that i could arrange the money accordingly.
    1. **Payment Window Epic** – Students that have chosen a living-unit to rent, shall be able to go inside a 'Payment' page which includes secure payment methods.

User-Stories:

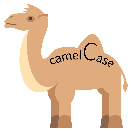
* + - 1. An order page will be displayed containing the condo's information that the user has selected and the final price. The user will enter all personal details and payment methods. If the order details are correct, the order will be saved in the system and sent to the tenant, otherwise an error message will be displayed.
      2. As a student, I can choose a payment method for the living unit such as PayPal, visa, credit card etc., so that it will be easier for me to rent a living unit.
      3. As a user, I could enter my payment in a well-managed interactive window, so I can pay and view payment details.
    1. **Website Support Epic** – Users that visit or those who signed-up to the Website shall be able to see notifications when the website is down, report bugs and have Customer support by Admins.

User-Stories:

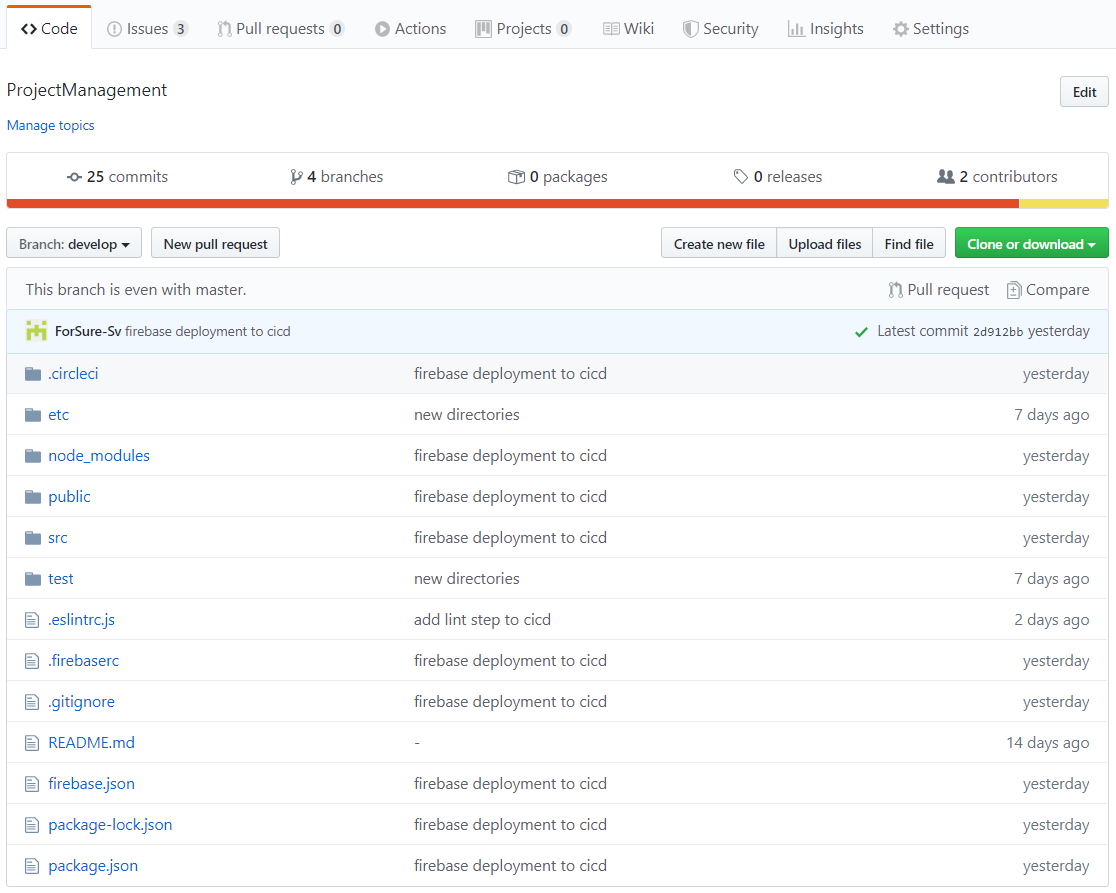
* + - 1. as a User, Unless the website is non‐operational, the system shall present to users a notification informing the users that the website is unavailable.
      2. as an admin, I want to be able to view all tickets properly as a table with the ability to tag each case progress and reply accordingly, in order to give proper customer support.
      3. as the Owner , I want to review submitted tickets to the website handle them accordingly, in order to give support to its users.
      4. as a Owner, I want an account with full permissions so that I can all of the website features and the ability to review tickets, so that I can manage the website more easily and see its progress.
      5. As a user, I can report bugs, in order to update the site managers.
    1. **Backend Management Epic** – Whole information shall be saved in databases that have to be updated in a regular basis while the system is online.

User-Stories:

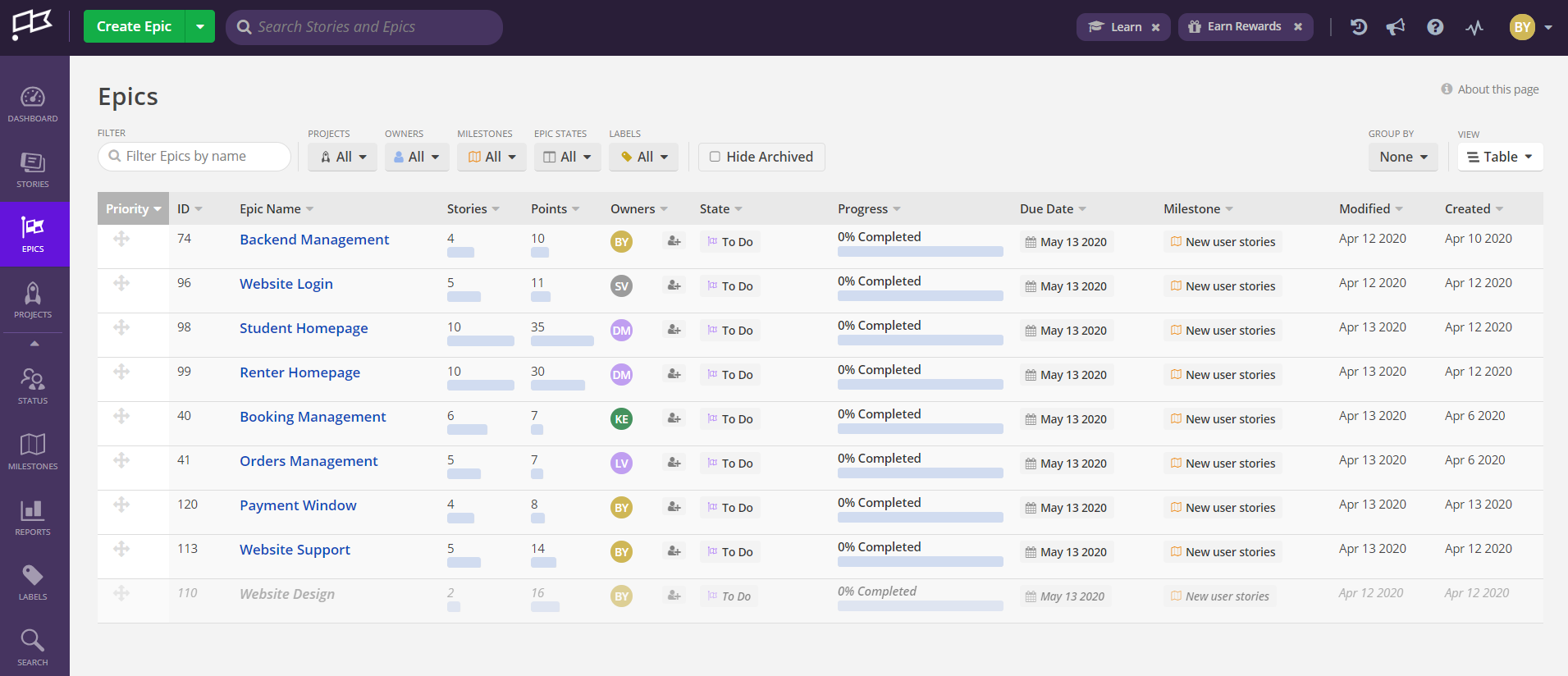
* + - 1. As a system admin, I want to backup all databases after a certain amount of time, in order to ensure a fail-safe for the system in case of a corruption in the database.
      2. As a system admin, I want to update the Living-Units Database after renter adds/removes a unit to keep the information displayed is up to date.
      3. As a system admin, I want to update the Users Database after sign-up process for the user to be able to sign-in the website later / after log-out.
      4. As a system admin, I want to update the Living-Units Database after a successful reservation to remove the living-unit from the units that are still displaying for the students.
* Note: Both Epics & User-Stories can be found in our [ClubHouse](https://app.clubhouse.io/scekdsl/epics)!
  1. Non-Functional Requirements
     1. **Security**
        1. Database protection – As we are dealing with client's personal information such as names, addresses, credit cards etc. the database should be secured to prevent from these data to leak.
        2. Users must change the permanent login password they receive after password change process immediately after successful login.
        3. Password length shall be of 6-10 characters including letters & numbers.
     2. **Performance**
        1. Fast I/O – To further user personal experience the system should be able to provide service quickly for the users, Under one second per interaction.
     3. **Scalability**
        1. System capacity – The system should be able to work with 500 users simultaneously without crashing.
     4. **Efficiency**
        1. The system shall be able to handle the entry of orders by customers at a minimum rate of 10 per second.
     5. **Durability**
        1. All databases must be backed up locally to prevent loss of data in case of malicious attack or fatal system crash.
        2. The system shall be stored on a well-known & reliable cloud platform to ensure as much as possible its durability.
     6. **Availability**
        1. The Online Payment System shall achieve 100 hours MTBF (mean time between failure).
        2. The online payment system shall be available 24 hours a day, 7 days a week.
        3. Unless the website is non‐operational, the system shall present a user with notification informing them that the website is unavailable.
  2. Test-Cases

1. **Project Infrastructure**
   1. Convention – In this project we will use the 'camelCase' Convention.

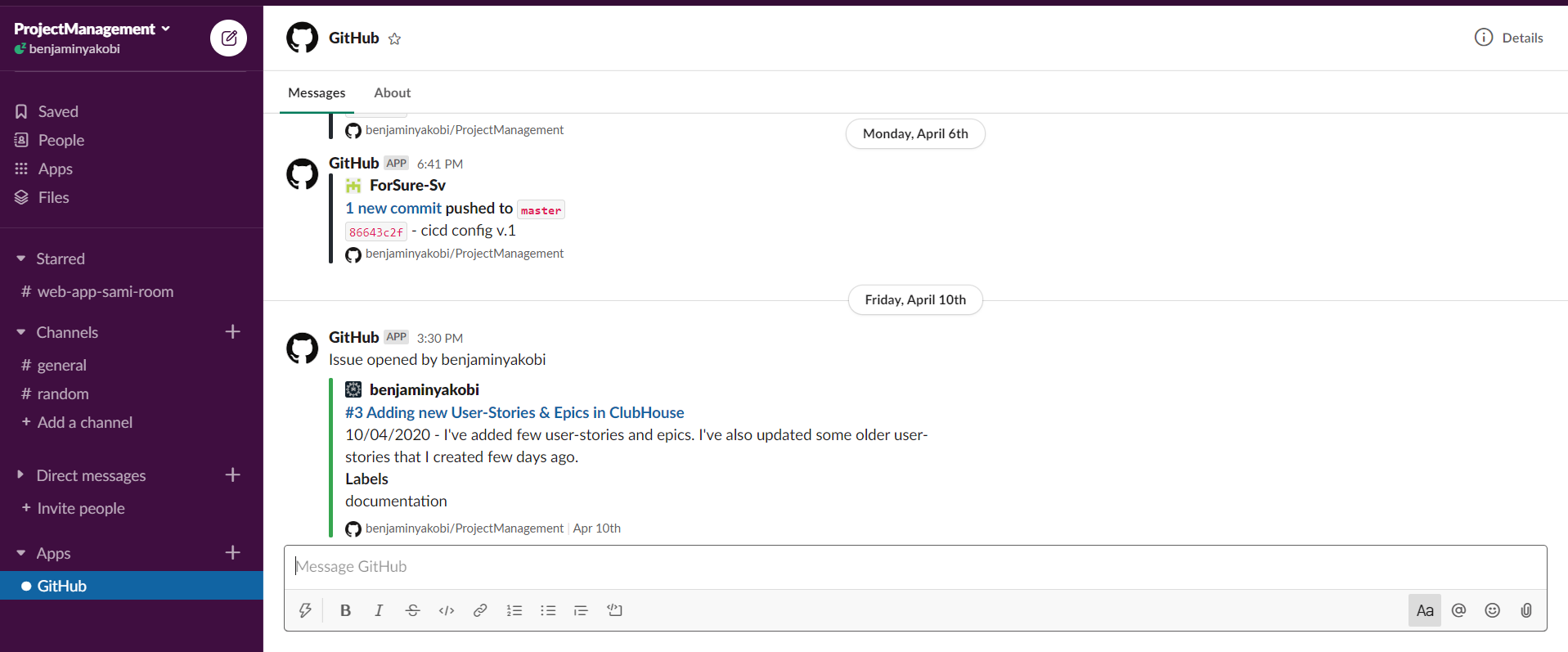
* 1. [GitHub Repository](https://github.com/benjaminyakobi/ProjectManagement) & Screenshot:



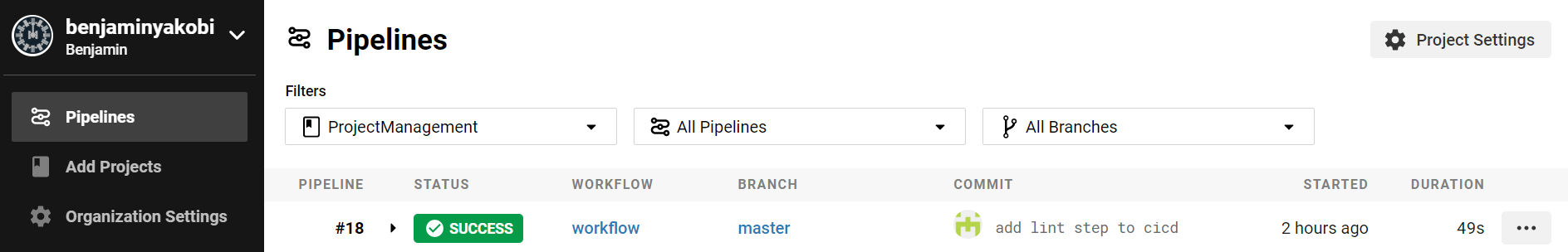
* 1. [Clubhouse Invitation](https://app.clubhouse.io/invite-link/5e7b729b-c807-4081-a089-08a3f44e208d) & Screenshot:



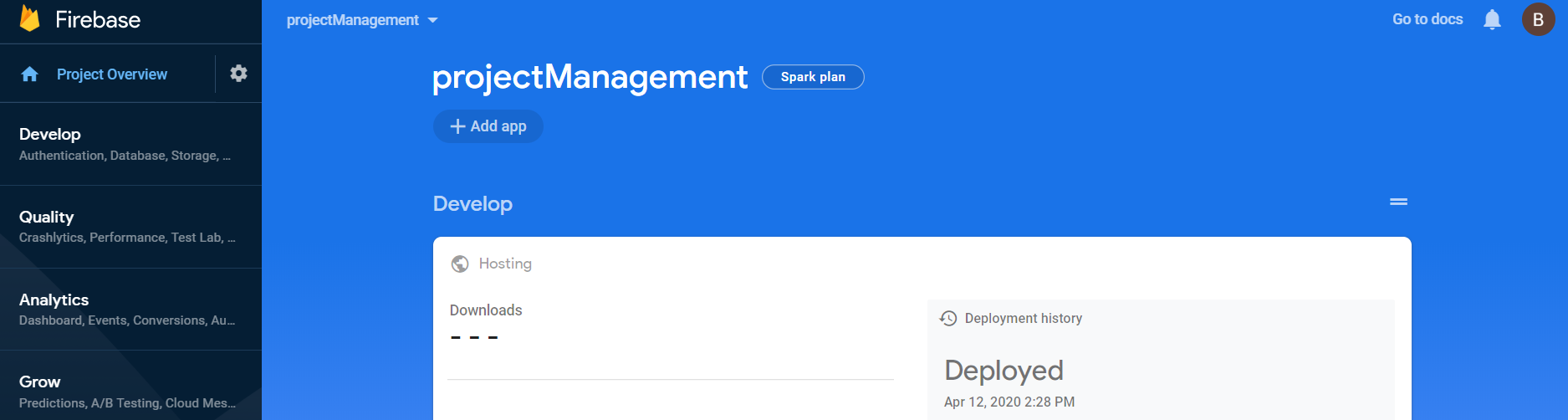
* 1. [Slack Invitation](https://join.slack.com/t/projectmanage-aem8080/shared_invite/zt-da9gfu1c-gmADLQBcDRu6s~aWyvI1aA) & Screenshot:



* 1. [CircleCi](https://app.circleci.com/pipelines/github/benjaminyakobi/ProjectManagement) & Screenshot:



* 1. [Firebase](https://console.firebase.google.com/u/0/project/projectmanagement-612b8/overview) & Screenshot:



* + In this project we will use Firebase as Cloud service to store & run the project.
  + In this project we will use Firebase as our DB to save our data for the next sprints.