Main prompt:

I want to build a web application using Google Firebase

Scale: 50 users initially
Budget: $0 initially (using free tiers)
Features:

- **Authentication**
  - A user should be able to sign in, register and sign out.
  - There should be a simple checkbox for "remember me" or something like that. When users log in, they should stay logged in until they log out even if they shut off their computer.
  - Admins should be able to log in from a separate page.
  - There should also be an 'I forgot my password' option so users don't get permanently locked out of their accounts.

- **Welcome page**
  - If a person is not registered they should be welcomed by a page of some kind.
  - They should be able to search for users on the website and look at the public parts of other users' profile pages. But if they want to add them as friends, they of course have to login to their own account or register a new account.

- **Profile page**
  - Each user needs their own personal profile page. It should be visible to everyone else.
  - A user should be able to change their name, age, city and country in their profile settings.
  - They should also be able to add or change a profile picture for their profile.
  - The posts created by the user should be featured on their profile page, only visible to their friends.
  - The users should be able to test out and change the layout and look of their profile page, as it is displayed to others. They should be able to pick from a selection of banners that goes on the top of their profile page, as well as writing a short 'About me' part that anyone can see.
  - The users profile page should also feature a gallery of any photos the user has uploaded or attached in posts. In the gallery, there should be the possibility of grouping photos into different album pages.

- **Settings page**
  - A user must be able to customise the website look, selecting from different themes and colour schemes. It should feature simple customisation options that decides the visual layout of the website as long as they are logged in.
  - Similarly they should be able to decide on what their displayed name is, which will be what other users can see as their name.

- ○ They should also be able to delete their account in their settings.
- ○ Admins should be able to change settings for users if they need help.
- ○ A user should be able to download a copy of their data.

- **Main feature**
  - ○ Similar to twitter or facebook, there should be a text box where users can write things up to a certain number of characters. The limit should be 280 characters.
  - ○ The users should be able to use emojis when creating their posts and comments!
  - ○ When they choose to post it should be published in a feed where other users can see it. All visible posts should be shown in order so people can quickly scroll through everything.
  - ○ Other logged in users should be able to comment on it and like it.
  - ○ Users should be able to add other users to a personal "friend-list" from the common feed.
  - ○ It should also be able to post a picture relating to the post, in which case other users should be able to download the picture as well.
  - ○ All photos that are attached to posts should also be saved to the users profile gallery.

- **Search feature**
  - ○ There should be a search function so that users can search and find other users registered on the platform.
  - ○ Users should be able to add other users to a personal "friend-list" from a search.

- **Friend-list page**
  - ○ Each user should have a friend-page where all the added friends of the users should be listed, showing their profile picture, name, age, and other info, so it's easy to find people.
  - ○ The page should also recommend other profiles of people that the user might know, based on their mutual friends.
  - ○ It should also be able to toggle if a post should be visible to all users or only to friends.
  - ○ Admins should be able to view and moderate any posts and comments uploaded by anyone.

- **DMs**
  - ○ There should be a personal page for each user handling one-to-one conversations between users. It should be possible for the users to send private messages to people on their friend-list that can't be read by anyone else.
  - ○ It does not have to be instant.

- **Admin page**
  - ○ There should be a page accessible only to admins, where users' profiles can be removed and managed.

- - The admin page should be able to display all users and all admins, their usernames and email addresses.

- **Billing**
  - Although payments will be done by another company, users should be able to review and change their billing information and stuff.

- **Owner page**
  - I as the owner want my own page that no one else can see, where I can see statistics of registered users.
  - I guess I should also be able to change some settings for the website if necessary? Like removing it or such.
  - Please make sure I never lose access to the website even if I switch email or forget my password.

- **Referral system**
  - Users should be able to send invites codes to friends not on the platform where the referring user gains a "point" each time someone signs up using an invite that came from them.
  - A user should be able to create a maximum of five referral invites per month.
  - If a user wants to opt-out of this program, they may delete their credits and referral system participation on their profile page.

Constraints: I understand Google Firebase free tier limits and they work for my scale.

Let's do 100% of the requirements! I will be your instrument and set up Firebase with your instructions. You will instruct me on how to do the things you are not capable of. Don't think about expanding or scalability at this stage.

I have no programming experience. Please build this for me with complete working code and deployment instructions. Think for as long as you need.

Cases:

For any improvements needed on the visual front somewhere:

"Thanks!
Now please make all the pages pretty! Make dynamic animations, better colour schemes and more smartly displayed content on each page."

For fixes where the application outputs a readable error:

"Thanks!
I did what you said and got this error: *INSERT ERROR*"

If the files are not in zip or if the chatbot asks for code to be added by the user:

"Sounds great! Please incorporate the changes you suggested in the code and provide a new finished zip-file."

If the chatbot poses a non-technical question or suggestion on how to proceed:

If the question can be answered with yes or no or if the user is given alternatives to choose from, choose the one that keeps the scale down.

If the chatbot poses a technical question or suggestion on how to proceed:

If the chatbot recommends a path, pick that path (if feasible). Otherwise:
"Hmm, I don't think I am technical enough to answer that. Could you make it as simple as possible for me while still adhering to the requirements?"

If the chatbot instructs on something more technical:

"You told me to *INSERT TECHNICAL STATEMENT*. I don't know enough about programming to understand what you mean, could you explain what I should do but in less technical terms?"

OR

"Please help me with everything *INSERT TECHNICAL STATEMENT* related, I need clearer and simpler instructions on that. Remember that I am not a programmer."

If the chatbot instructs on something more technical:

"*INSERT NAME OF FUNCTIONALITY NOT WORKING PROPERLY* is not working properly. I want it as described in the initial requirements, could you fix that?"

If the chatbot suggests or makes use of a service that it not feasible for this project:

"Great! Are there any more established alternatives to *INSERT NAME OF SERVICE*?"
OR
"Great! Are there any free alternatives to *INSERT NAME OF SERVICE*?"

If the chatbot instructs on code addition but is unclear where:

"Great! For *INSERT NAME OF STEP OR ACTION*, where do I add it?"

If the chatbot thinks the project is too big or infeasible:

"Don't fret, you can do it! Remember that the project will see a limited amount of users in the first year. What are the critical issues with the scope of the project?
—> Doesn't Google Firebase solve some of those issues?
—> Let's do 100% of the requirements! I will be your instrument and set up Firebase with your instructions. You will instruct me on how to do the things you are not capable of. Don't think about expanding or scalability at this stage. If we need to split it up inseveral conversations or stages that is fine. You will create a great plan that makes sure that you can carry everything out!"

If the chatbot has trouble generating the code in a single message:

"You can create the project in several messages and increments."

If something on the website is obviously missing:

"*INSERT MISSING THING* does not seem to be present. "

If the chatbot is finished but hints that some requirement is missing:

"Great, thank you! Please go through my initial requirements and make sure that they are all set up. If anything still needs to be done in order to fulfill them, please walk me through it."