# Documentation for Angular app - Amigo
# By Adhiraj Singh

**Idea -** Friendship tracking web app that allows you to keep track of friends' birthdays and their interests.

**Motivation -** Sometimes in our busy lives it is hard to remember things about your friends and family in the day to day lives and there should be a focused and simple web app that allows me to collate all the information in one place. This will help make it easier to plan birthdays in advance, have ready to implement gift ideas and just make caring about friends a game-like experience.

**Execution -**
1) Framework - Angular 14
2) Styling - CSS
3) Pages - v1 - home page (welcome + adding friends) , My dashboard with all friends represented by cards
4) Storage - Cannot afford cloud based storage - due to simple data will use end user local browser storage
5) Hosting - Netlify/vercel
6) Testing - Cypress

**Themes -**
1) Vibes - friendly, easy, straightforward, encouraging,
2) Colors - modern , sleek, 2 color, comfort.
3) Rounded and simple

**Homepage -**
The homepage will have a simple welcome banner , a button to add a friend, and navigation bar for the dashboard and about me page

**Add friend -**
1) Form animates in
2) 3-4 entries  - trial list - name/nick name, birthday, interests, gift ideas
3) Submit -> info saved -> card created -> added to dashboard
   a) Loop through added friends
   b) Load each info into new card
   c) Display card

**Dashboard -** Cards listing friends and their information
- Reminder 1 week before birthdays

- XP bar showing progress each time you report a successful bday remembrance or interest list grows
- Cards listed out 4 per row
- Animate on hover - goes to nothing to reduce complexity - could expand or go to a new page with all the details?

**Timeline -**
1) Homepage - 2 days - **2 days**
2) Dashboard - 4 days - **6 days**
   a) Cards - 2 days - **4 days**
   b) XP bar - 2 days - **2 days**

**Misc progress reports -**

**Obstacle 1 -** I wanted to add information to the local storage. But making users add id numbers is immersion breaking, plus complicating the code with my own id generator would make iteration difficult later on to fetch the data.

So at the moment only the individual data json objects were being added upon form submission to the local storage.

However I came up with an idea.

1) Add an empty array to the local storage as soon as the component is created
2) When submitting information -
   a) Call array
   b) Push latest value
   c) Send latest array json stringified to the local storage
   d) It works!
   e) There must be an easier way to do this. But I was keeping it as simple as possible.

**Obstacle 2 -** The form was being submitted even with empty fields. I wanted to make at least the name a must fill field. Despite me making the input required it was not working. After reading the documentation -
1) It could be a browser issue
2) It could be a tag issue
3) It could be the wrong type of submit button

None of these fixes worked.

I added ngNativeValidate to the form header. It worked!

**Obstacle 3 -** What a silly mistake, that empty array initialization in step one would override all the local storage - so I fixed it by checking if it exists first, and only then adding the empty array to the local storage

**Obstacle 4 -**
The cards on the friends dashboard simply will not show the details
   1) Changing array type did not work
   2) Converting the data did not work
   3) The data was just fine when I console logged it.

So I realized I am iterating through an object of objects here.

   1) Each ngfor iteration gives me access to current object, but not its inner workings
   2) So a second loop is needed to iterate through that specific object

Voila, all fixed.

However, the data will not show up properly.

   1) Added key value piping due it being an object

Data showing in alphabetical order which ruins the experience

   1) Added a return 0 function to the keyvalue pairing ngfor statement, and it is all good now

Must style the cards and make them look good next
Plus work on using length of friends for an xp bar

**Obstacle 5 -**
Figuring out the closest birthday
The idea was to find the closest birthday coming up no matter when you made that friend card. So I iterated through all birthdays if they chose to add it to the calendar and do some math to find the shortest distance between birthdays and today's date.

My concerns were -
   ● what if they put in the year they were born ? The calculations would go haywire
   ● What if their birthday is gone already but it is the latest birthday coming up?

Final solution
   1) Iterate through birthdays

2) Variable for current shortest difference in months and days update if current iteration is closer than previous difference in months
3) Stringify and extract only month and day
4) Stringify today's month and day
5) To find months left -
   a) If birthday month>todays month
      i) Subtract birthday minus today's month
      ii) Display month with corresponding friends name
   b) If birthday is in the same month
      i) If birthday day > today's day
         (1) Birthday minus today's day
         (2) Display days left

**Feature ideas -**

☑ ~~Go to roster button pops up once you click on save friend, following same line of animation as the form itself for consistency, ofcourse, the option is available on the menu bar too~~

☑ ~~Birthday countdown - somehow take the birthdate they input and set up a countdown timer~~
   ☑ ~~Take in a text date + take in a proper date format bday date. DONE~~
   ☑ ~~Display only text date but use the date input for calculations DONE~~
      ☑ ~~Iterate through all birthdays and find the latest birthday?~~
      ☑ ~~Display that birthday~~

☑ ~~XP bar for each friend you add. Either one single total xp bar or one for each friend. The XP will go up because of the following factors - number of likes added, birthdays added, nicknames given~~

# AMIGO V1.0 Final feature list

**Components -**
1) Home page
   a) Title → Welcome, Friends
   b) Logo → Adobe express custom logo
   c) Add friend button → loads form
      i) Form has the following input slots
         (1) Name (required)
         (2) Birthday (text)
         (3) Add birthday to calendar
         (4) Likes and interests
      ii) Form submit → loads navigation to friends roster
      iii) Form submit → save data to local storage
      iv) Form submit → resets inputs
   d) Go to roster button → Navigation to friends roster

2) Friends Roster
   a) Title → My Friends
   b) XP number
      i) On hover reveals how xp value is calculated
   c) Level number
   d) Information about xp calculation on xp number hover
   e) Latest birthday coming up with how many months or days left
   f) Friends cards
      i) Iterate through local storage and display cards
      ii) Cards have white glow on hover
      iii) Cards always up to date using *ngfor on component creation

**Color palette -**

Colors used -
1) Yellow - **#F5D33C**
2) White
3) Black - **#1a1a1d**
4) Green - **rgb(14, 184, 42)**

# Welcome, friend.

Add a buddy

My Friends

## Please input their details!

**Name / Nick Name**

Name

**Birthday**

Birthday

**Add Birthday to Calender (Optional)**

dd-mm-yyyy

**What do they like? (Food, Animes, hobbies etc)**

sushi, action figures, knitting....

Save

sushi, action figures, knitting....

Save

View all Friends

# My Friends

Your XP: 500$^?$     Level: 1

hehehe's Birthday in 3 Days

Your XP: 500$^?$     Level: 1

The XP is calculated with weightages of number of friends and how many things you know about them

hehehe's Birthday in 3 Days

**Test1's Birthday in 7 Days**

| Name: Test1 | Name: Test2 | Name: Test3 |
| Birthday: | Birthday: 23rd November | Birthday: 15th May |
| Likes: Food | Likes: | Likes: Sushi, dancing |

-----------------------------------End of V1.0 documentation—-----------------------------------

## First User testing round

1) Add a link to go back to the homepage on the friends roster page - Fixed
2) The superscript question mark showing up without any friends present - Fixed
3) Mobile responsiveness not complete yet - adjust css based on screen size
4) If the birthday is in the same month - the days are correct but the name does not change

—----------------------------------------------------------------------------------------------------------

# AMIGO V2.0

So just 3 days after launching V1, I was overflowing with ideas and I knew something was missing. Here is everything that was on my mind

1) The border color of the cards has to be the same color as the user input
2) Need to make the cards pop more

This version update was less about the functionalities and more about user experience and interface

So here is how I implemented it

**Coloured borders -**
1) Add an input tag of type color
2) OnInit adds the color separately to the color list.
   a) Why separately?
   b) This is because my information iteration happens in the container within the card so to access ng style would not work on an internal element
3) Upon iteration use an ngfor index to call the color array and add border styling
4) The array is always using the right index as I have saved them all in the right order

It looks beautiful

**Surprise element -** Making the website **POP**

The original idea for this website came when I was playing pokemon. The collecting and rostering of pokemon gave such a fun feeling to going around waiting for strange pokemon to pop up.

Life felt like that, in the department of making friends. Weaknesses, strengths, leveling up, creating a bond. Of course, human friends are more complicated than pocket monsters. But this should be a start.

So I had the stats, I had levels, I had information. We were missing the SPRITES.

So after 3 hours of research I found an API by an amazing artist named Ashley Seo and the api was provided by Dicebear.

The artist's Website is this - http://www.ashleyseo.com/. A big thank you to her for her amazing work.

I was able to call the api, randomize the hair, skin, expression, smile with some coding (arrays, calculating mods, indexing, slicing, ngfor) and even figured out how to make sure the color of the sprite was the same as the color the user picked for their friend!

The output looked amazing and 5/5 of my test group approved the design. I will show the final outputs in the final user interface collection

**Obstacle 1 -**
    1) The color in the color input list is stored as a #000000 code. The api, despite having an inbuilt regex, is not accepting it during iteration.
        a) Fixed it by slicing the string using the slice pipe and removing the # symbol. All works again!

**Obstacle 2 -**
    1) The sprite icon keeps moving as it sets its css relative. This occurs due to the auto adjusting card size based on input length. This means the x and y values adjust relative to the new bottom of the card.
        a) Fixed it by setting the position centered and absolute. The icon hovers above and sticks with the card.

**Debugging -**
    1) Need default color to be displayed in case user picks no vibe color as it is throwing ngfor for a spin - fixed
    2) There seems to be something wrong with the same month birthday calculations , test more and fix -  in user testing

**Final Frames**
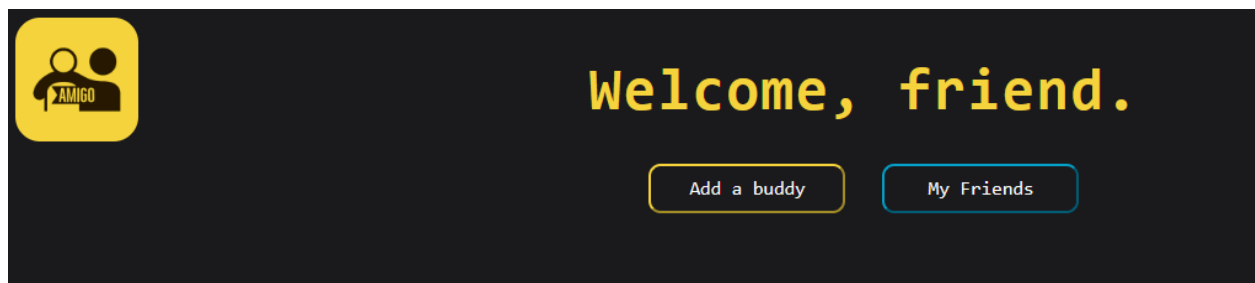
# Welcome, friend.

Add a buddy    My Friends

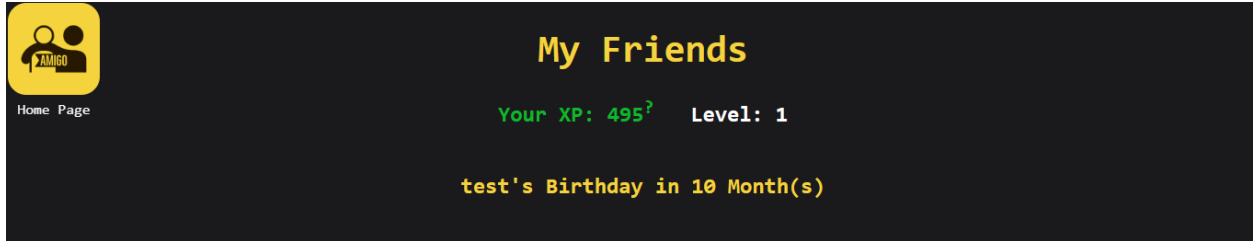## Please input their details!

### Name / Nick Name

Name

### Birthday

Birthday

### Add Birthday to Calender (Optional)

dd-mm-yyyy

### What is the colour of their vibe?

### What do they like? (Food, Animes, hobbies etc)

sushi, action figures, knitting....

Save

HERE COMES THE SURPRISE

test's Birthday in 10 Month(s)

Name: ef

Birthday:

Likes:

Name: test

Birthday:

Likes:
ew,ew,,w,w,e,wee,we

Name: sdsf

Birthday:

Likes:

Name: Kaaveri Singh

Birthday:

Likes:
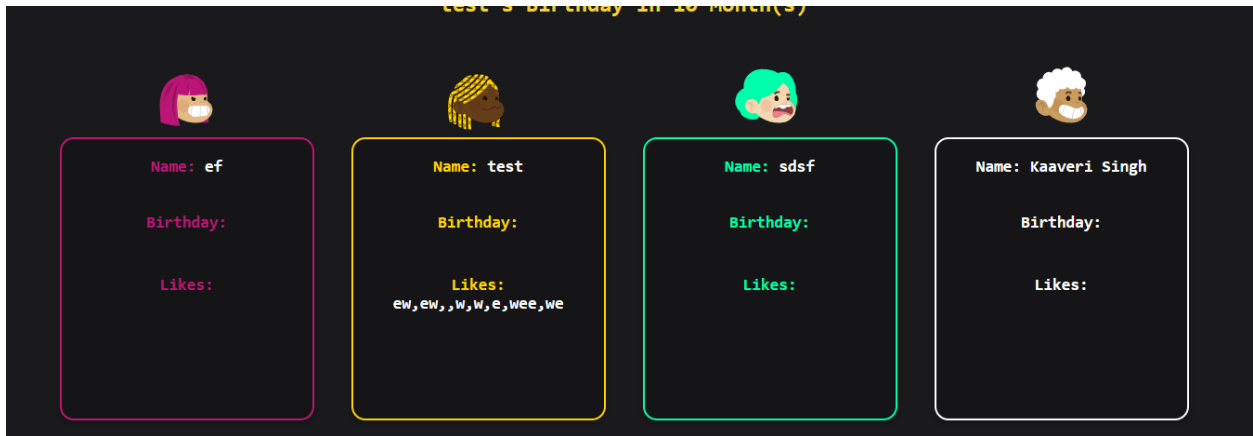
---------------------------------------------------------END OF V2---------------------------------------------------

**Testing feedback**

**Ideas for V3**
1) UI is done, anymore will cause eye strain
2) Suggesting gift ideas
3) Suggested activity ideas
4) More optional form inputs - first met, years known, best memory