THE UNIVERSITY OF SYDNEY

School of Architecture,
Design & Planning

FAMILYSIM MAGICMIRROR

DECO4200

12 NOVEMBER 2021

Group KHMC

Members Caitlyn Burton
Harrison Khannah
Kim Truong
Miguel Tamondong

Table of Contents

1. Introduction

Background

The Keeping in Touch brief demonstrates the struggle for the elderly to communicate with one another during the COVID-19 pandemic. This social isolation can lead to a variety of detrimental effects.

For the elderly, the risk of cognitive decline increases severely when social connections and activity are limited (Nicholson, 2012). Through the COVID-19 pandemic, the popularity of applications such as like Zoom have surged as methods for keeping in touch with friends and family in the absence of physical interactions. However for older adults their needs and wants are often neglected when it comes to developing new technologies (Pedell et al, 2013).

The problem brief highlights social isolation as a problem especially for the elderly living in times such as the height of the COVID-19 pandemic. Varioius online online tools have popularised as a result of the social and geographical isolation caused however according to our primary research, many of these tools are difficult to use among the elderly and therefore not utilised by them. This emphasies the need for an elderly-friendly online tool to promote communication to address the lack of social interaction.

The FamilySim Magic Mirror was developed to address a lack of social connection particularly for the elderly by providing them with a device which is unintrusive and can naturally integrate into their daily lives (as a mirror is a device which is walked past daily.

2. Overview of the Design Process

Double Diamond Design Approach

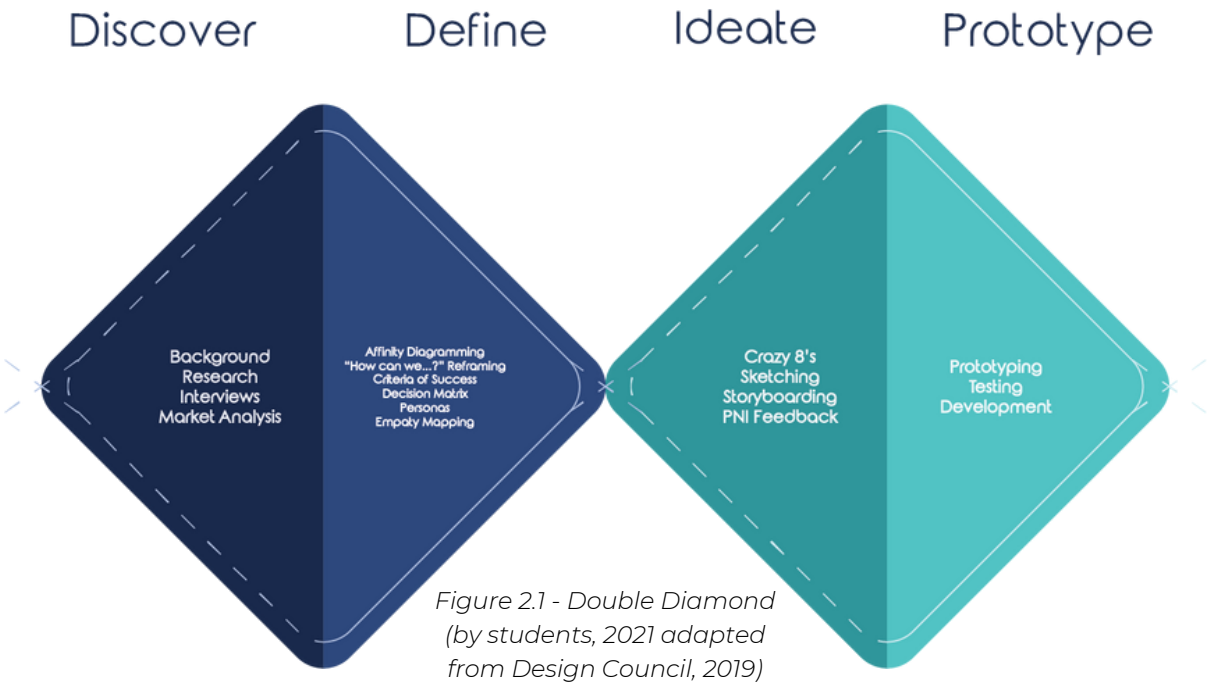


Figure 2.1 - Double Diamond (by students, 2021 adapted from Design Council, 2019)

Our team followed the traditional design process: the Double Diamond Design Approach. Using a suite of UX design methods, we explored our areas of interest in the brief, refined the scope of the project in our problem definition, ideated from the differently framed problem definitions, and prototyped and tested our final solution to satisfaction.

Discover

Background Research

In exploring different specialisations of the brief, our team underwent secondary research into journals on several topics. The insights derived from this research lead to our criteria of success.

Interviews

Using interviews as a tool to discern background information about the problem area (Tomitsch et al., 2018), We discovered that older users are not only unafraid of new technology but seek it as to stay in the loop.

Market Analysis

PNI analysis (Positive, Negative, Improvement - adapted from PMI analysis (de Bono, 1988)) was performed on six different social and mental stimulation apps, revealing a gap in communicative functionality and lack of accessibility for older users.

Define

Affinity Diagramming

Affinity diagramming allows extensive collaborative research to be collated and organised to reveal underlying user needs (Pernice, 2018) and provided the basis of our criteria for success

"How Can We...?" Reframing

Using the “How can we...?” reframing method, adapted from the “How might we” method, allows us to generate creative solutions by rethinking different issues our research has identified (Rosala, 2021).

Personas & Empathy Mapping

Personas provided the team with a clear vision of the users to ensure their needs are being met by the solution (Junior & Filgueiras, 2005). Empathy mapping was also undertaken in order to build a foundation of the user experience (PlaybookUX, 2019).

Criteria of Success/Decision Matrix

Using the user needs and wants identified from affinity diagramming, our team collated criteria for success. These criteria were set aside for evaluation of future concepts through the undertaking of a decision matrix that would fairly quantify how well each concept is able to address the issues that we had identified (Chang, 2015).

Ideate

Crazy 8's

Crazy 8’s is a rapid concept generation technique that favours quantity over quality (Chung, 2020), allowing a limited amount of time to conceptualise and illustrate 8 solutions.

Sketching

In order to help represent the ideas created in Crazy 8’s, we used sketches alongside the short descriptions of the concept to illustrate how each solution was intended to be interacted with or how it could work (Tomitsch et al, 2018). The team developed 32 sketches, one for each "crazy 8" concept.

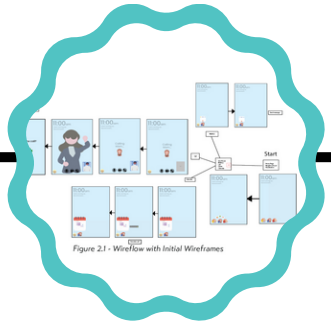
Storyboarding

Storyboarding is a visual technique that allows us to begin thinking about the holistic user journey, highlighting their pain points and demonstrating how solutions could work (Krause, 2018). Storyboarding further developed our concepts by illustrating the full user journey.

PNI Feedback

In the first workshop with the medical science students, we prepared a basic PNI (positive, negative, improvement) activity for them to complete for each of the storyboarded ideas. This feedback was discussed and taken into account alongside our decision matrix in determining the final concept and what key features from each idea we would continue forthwith.

Prototype (and Test)



Wireflow w/Wireframes

The first iteration of the concept to be subject to testing was a wireflow with accompanying wireframes. A wireflow demonstrates the flow of screens and information within the user interface when performing different tasks (Laubheimer, 2016).

Wizard of Oz Testing

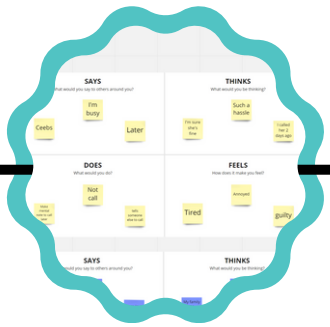
The Wizard of Oz method allows us to simulate how users would interact with a device interface and perform different tests before the interface is fully functional (Alce et al., 2015). This technique is especially useful for exploring how users interact with a physical device in the intended environment, concluding our solution is best in a public space in the home.

Card Sorting

Card sorting is a method that provokes test users to determine for themselves the most logical organisation of features and information in the system's navigation and architecture (Tomistch et al., 2018).

Figma Prototype (1st Iteration)

After determining the initial features, a figma prototype was developed to connect the screens depicted in the wireflow with interactive buttons that can be pressed in preparation for the testing activities in the medical science student workshop.



Empathy Mapping

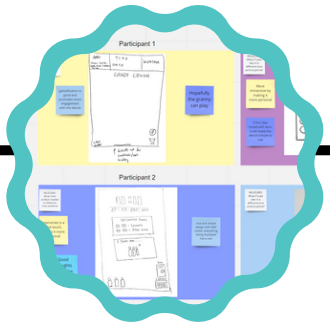
In order to perform group sketching, the workshop students underwent empathy mapping to reimagine themselves as the intended user persona in order to immerse them into the user's environment as completely as possible (Bland, 2020).

Group Sketch

Test users undergoing group sketching imagine themselves as the intended user persona and illustrate key features of the user interface that they believe would be the most essential using sketching (Gillbert, 2018). This method developed the UI to be more recognisable to existing devices and expectations in the next iteration.

Video Prototyping

Video prototyping demonstrates to test users what the physical environment and context of a device's intended use is, to walk the test participants through the user journey and receive feedback for the success in the implementation of the device (Tomitsch et al., 2018). This method revealed ambiguities in the neighbourhood metaphor and provided a basis for the next phase of testing.



Wildcards

The wildcard activity performed with the medical science students provoked them to redesign the final concept while focusing on a specific user need or criterion in order to explore how well the concept is addressing the full criteria of success (Situ, 2019). This activity suggested new features such as games and photo memories.

Figma Prototype (2nd Iteration)

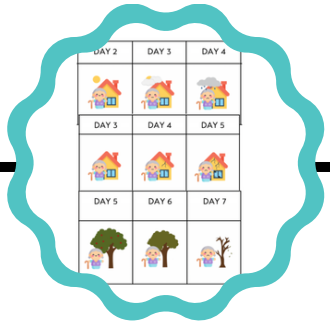
New features were added such as post-call feedback, user settings, different types of messages, games, photo memories, and different neighbourhood metaphors. The prototype is black and white, taking position and icon design into account.

Figma Prototype (2nd Iteration)

A/B Testing allows us to test two different versions of an interface to determine which is more intuitive or effective for the user (Kohavi & Longbotham, 2016). Conversation prompts demonstrated their value and were added to the next iteration.

Figma Prototype (3rd Iteration)

The final Figma prototype development has implemented all changes determined from the last two testing techniques: the new weather metaphor and call prompts, with the addition of colour into the design.

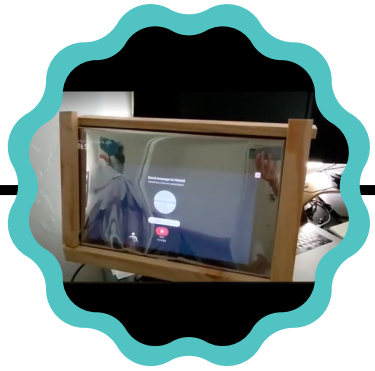


Multivariate Testing

Multivariate achieves the same purpose as A/B testing (determining the best version of a design) but with more options being able to be tested at the same time (Holzmann & Hutflesz, 2014). This method was used to test different metaphors for the neighbourhood visualisation such as weather, architectural degradation, emoticons, and trees with the weather being the most prominent invoker.

Software Development

The prototype was implemented into an independent system using a Linux-based Electron application, joined with a React layer on the Electron applet. Details of the software system architecture are detailed later in the report. This application has been developed throughout the semester with several adjustments to the interface, additions of animations and other details throughout the testing phase.



Physical Development

Physical development of the prototype had also been performed throughout the semester. Firstly, testing of the mirror film material was performed to determine how it could be best mounted onto a screen, what colours show best through the film, and what environments make the screen and user most visible. Furthermore, the full physical prototype was constructed after the software development had confidently been completed.

Figma Prototype (2nd Iteration)

The second iteration of the Figma prototype included the addition of several new features and adjustments to old features such as post-call feedback, user settings, different types of messages, games, photo memories, and different neighbourhood metaphors. The prototype is black and white, now taking position and icon design into account.

A/B Testing

A/B Testing allows us to test two different versions of an interface to determine which is more intuitive or effective for the user (Kohavi & Longbotham, 2016). The results of the testing proved the value of including prompt cards in the call feature to help facilitate the conversation as younger users revealed their discomfort in being pressured to have news to give in each call.

Multivariate Testing

Multivariate achieves the same purpose as A/B testing (determining the best version of a design) but with more options being able to be tested at the same time (Holzmann & Hutflesz, 2014). This method was used to test different metaphors for the neighbourhood visualisation such as weather, architectural degradation, emoticons, and trees with the weather being the most prominent invoker.

Figma Prototype (3rd Iteration)

The final Figma prototype development has implemented all changes determined from the last two testing techniques: the new weather metaphor and call prompts, with the addition of colour into the design.

Software Development

The prototype was implemented into an independent system using a Linux-based Electron application, joined with a React layer on the Electron applet. Details of the software system architecture are detailed later in the report. This application has been developed throughout the semester with several adjustments to the interface, additions of animations and other details throughout the testing phase.

Physical Development

Physical development of the prototype had also been performed throughout the semester. Firstly, testing of the mirror film material was performed to determine how it could be best mounted onto a screen, what colours show best through the film, and what environments make the screen and user most visible. Furthermore, the full physical prototype was constructed after the software development had confidently been completed.

Card Sorting

Card sorting is a method that provokes test users to determine for themselves the most logical organisation of features and information in the system's navigation and architecture (Tomistch et al., 2018). This method demonstrated what information users thought should be available at all times and was implemented in the next iteration.

Empathy Mapping

In order to perform group sketching, the workshop students underwent empathy mapping to reimagine themselves as the intended user persona in order to immerse them into the user's environment as completely as possible. This allows the students to not only provide test data unique to their interpretation but also under the consideration of the intended context of the device as well (Bland, 2020).

Group Sketching

Test users undergoing group sketching imagine themselves as the intended user persona using empathising methods and illustrates key features of the user interface that they believe would be most essential for the user persona using sketching (Gillbert, 2018). This method helped us develop our features in a way that was more recognisable to existing devices and interface expectations when developing the next iteration.

Wildcards

The wildcard activity performed with the medical science students provoked them to redesign the final concept while focusing on a specific user need or criterion in order to explore how well the concept is holistically addressing the criteria of success (Situ, 2019). This activity provided the next iteration with new features such as games and photo memories.

Video Prototyping

Video prototyping demonstrates to test users what the physical environment and context of a device's intended use is, to walk the test participants through the user journey and receive feedback for the success in the implementation of the device (Tomitsch et al., 2018). This method revealed ambiguities in the neighbourhood metaphor and provided a basis for the next phase of testing.

Mindmapping

Mindmapping allowed the workshop students to freely brainstorm, drawing on their own knowledge, expertise, and experiences without the usual constraints of our design's tests (Davies, 2010). This activity used the expertise of the medical science students in the physical limitations of older users to ensure our device was exploring accessible modes of interaction.

Sketching

In order to help represent the ideas created in Crazy 8's, we used sketches alongside the short descriptions of the concept to illustrate how each solution was intended to be interacted with or how it could work (Tomitsch et al, 2018). The team developed 32 sketches.

Storyboarding

Storyboarding is a visual technique that allows us to begin thinking about the holistic user journey, highlighting their pain points and demonstrating how solutions could work (Krause, 2018). The team used storyboarding to further develop the three focused concepts in order to exhibit the mode of interactions and intended use case for users.

PNI Feedback

In the first workshop with the medical science students, we prepared a basic PNI (positive, negative, improvement) activity for them to complete for each of the storyboarded ideas. This feedback was discussed and taken into account alongside our decision matrix in determining the final concept and what key features from each idea we would continue forthwith.

Prototype (and Test)

Wireflow with Wireframes

The first iteration of the concept to be subject to testing was a wireflow with accompanying wireframes. A wireflow demonstrates the flow of screens and information within the user interface when performing different tasks (Laubheimer, 2016). This version of the concept introduced the metaphorical neighbourhood, calendar scheduling and calling features.

Wizard of Oz Testing

The Wizard of Oz method allows us to simulate how users would interact with a device interface and perform different tests before the interface is fully functional (Alce et al., 2015). This technique is especially useful for exploring how users interact with a physical device in different environments and ultimately determining the intended device location should be in a more public space in the home along with other insights.

Figma Prototype (1st Iteration)

After determining the initial features, a figma prototype was developed to connect the screens depicted in the wireflow with interactive buttons that can be pressed in preparation for the testing activities in the medical science student workshop. This first iteration of the Figma prototype provided the base for video prototyping later on.

Market Analysis

PNI analysis (Positive, Negative, Improvement - adapted from PMI analysis (de Bono, 1988)), on six different social and mental stimulation apps we confucted. With a gap in communicative functionality and lack of accessibility for older users, we were able to understand what our solution needed to focus on.

Define

Affinity Diagramming

Affinity diagramming allows extensive collaborative research to be collated and organised to reveal underlying user needs (Pernice, 2018). The collation of our research in the Discover phase revealed four main themes: Accessibility & Learnability, Ability to Connect with Others, Mental & Physical Stimulation & Engagement, and Experiencing Genuine Interpersonal Connection.

"How Can We...?" Reframing

Using the “How can we...?” reframing method, adapted from the “How might we” method, allows us to generate creative solutions by rethinking different issues our research has identified (Rosala, 2021). Using this method, the team were able to reframe the problem area into three different directions that the ideation phase was to focus on.

Criteria of Success/Decision Matrix

Using the user needs and wants identified from affinity diagramming, our team collated criteria for success. These criteria were set aside for evaluation of future concepts through the undertaking of a decision matrix that would fairly quantify how well each concept is able to address the issues that we had identified (Chang, 2015).

Personas & Empathy Mapping

Personas provided the team with a clear vision of the users to keep in mind as we undergo the design process to ensure their needs are being met by the solution (Junior & Filgueiras, 2005). Additionally empathy mapping was also undertaken in order to build a foundation of the user experience to aid the construction of the personas (PlaybookUX, 2019).

Ideate

Crazy 8's

Crazy 8’s is a rapid concept generation technique that favours quantity over quality (Chung, 2020). Group members are given a short amount of time to conceptualise a different solution and illustrate it before they must move to the next idea. This technique provided the team with 32 ideas to informally vote on and subject to a decision matrix against our criteria of success.

3. Core Functionality

Home

The MagicMirror Family Sim has 3 main capabilities, most of which is centred around scheduling and contacting specific family members and loved ones. The experience is personalised around the users based on the interactions they make using this device.

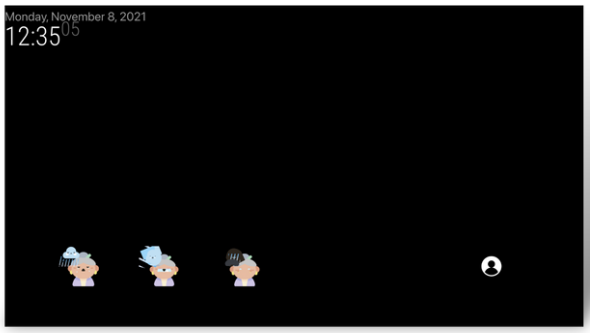


Figure 3.1 - Starting Screen

Users can interact with their family members through the home screen in three distinct ways:

1. Making an impromptu call
2. Checking their calendar, and scheduling a call
3. Sending them a message (via voice or text)

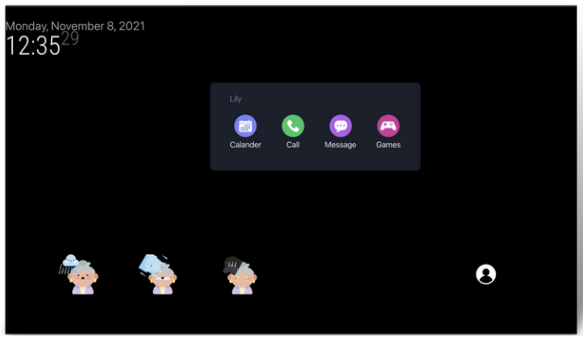


Figure 3.2 - Menu: Accessible by selecting a user's avatar

Calling Another User

The process of calling another user emulates the process as it exists on a mobile device, pressing the call button and being presented with the same functions as if you were on FaceTime etc. Here a user can toggle their camera and their microphone.

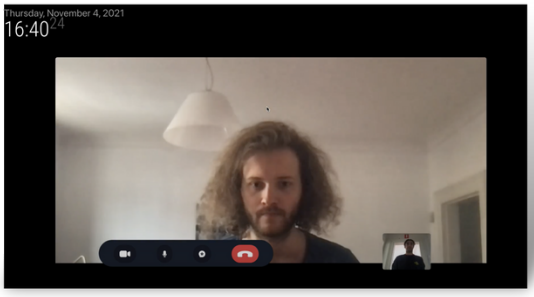


Figure 3.3 - Video Call

MagicMirror aids in facilitating conversation between family by providing "prompts" that display different questions or topics that users can answer to aid in the flow of conversation. These call prompts take up the middle of the screen and provide suggestions of conversation topics should the conversation die down.

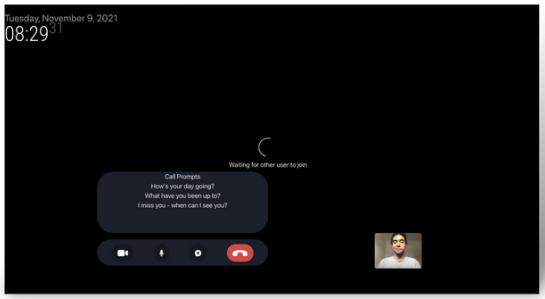


Figure 3.4 - Call Prompts

The post-call rating system is similar to similar modern call applications but conveys the quality of the conversation rather than the quality of the technical experience. A user is presented with 5 reactions, which can be used to simply describe their experience in their call with this person.

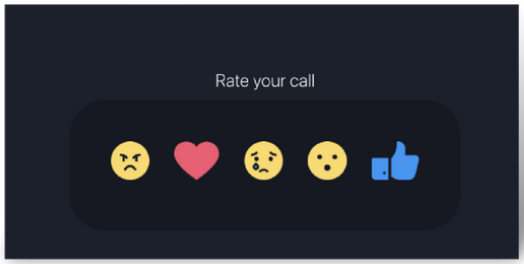


Figure 3.5 - Post-call Feedback

This information contributes to the visualised mood value of users' avatars, depicting if the relationship between the user and the caller has been improved, maintained, or worsened.

When positive interactions are recorded, sunny, happy weather is displayed

Figure 3.6 - Happy Avatar

When negative interactions (or a period of time occurs without positive interaction), gloomier stormy weather displays).

Figure 3.7 - Sad Avatar

Checking Another User's Calendar

A user's calendar can be viewed when the connected user has permitted you to look at it in their own privacy settings. From here a user can view when their connected user may be busy, and also schedule a time for them to call. The main purpose of this page is to allow for users to find a time when they and their connected user is free, so that they can set a time to call.

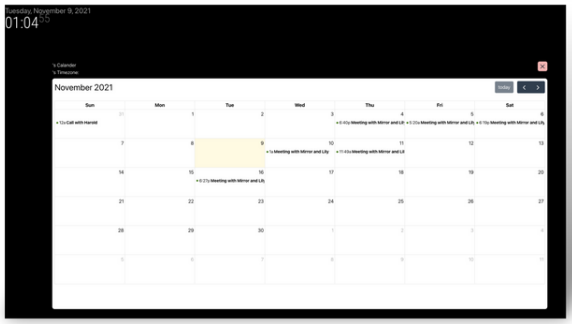


Figure 3.8 - Other User's Calendar

This is an alternative to calling out of the blue. From this page users can schedule a call by clicking on a date that is free, and interacting with the popup that confirms the specific time of the call. By scheduling a call, they will now receive notifications when the time of their call draws near.

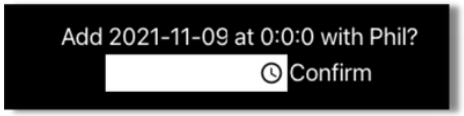


Figure 3.9 - Scheduling Call Confirmation Prompt

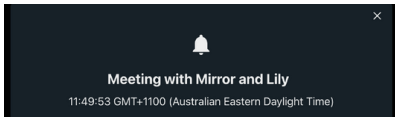


Figure 3.10 - Call Reminder Notification

Sending a Message to Another User

A user can choose from two different types of messages; a text message and a voice message, to send to their connected user.

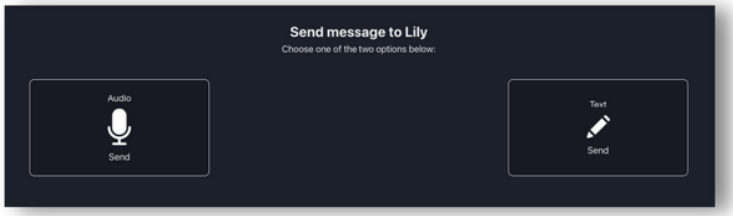


Figure 3.11 - Message Format Options

Voice Messaging

Operating the voice message section is very similar to operating applications such as Voice Memos. They simply:

1. *Press the mic button to record (see Figure 3.12)*
2. *Press the green button to send the message (see Figure 3.13)*
3. *View message success popup (see Figure 3.13)*

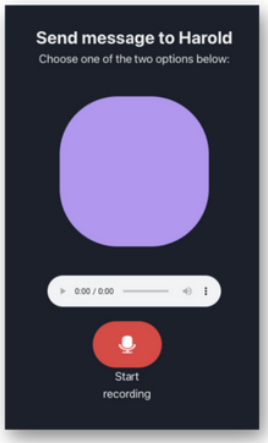


Figure 3.12 - Begin Voice Recording

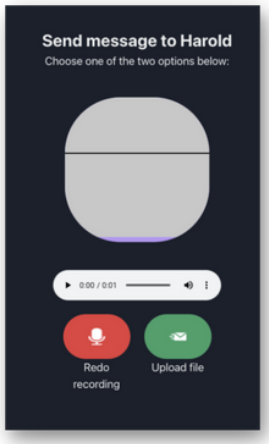


Figure 3.13 - Voice Recording in Progress

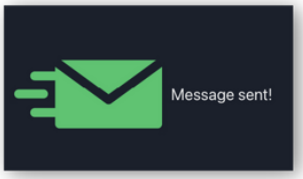


Figure 3.14 - Sent Message Confirmation

Text Messaging

The text messaging option provides users with different preset greetings to send their connected users or they may type their own with the bottom option.

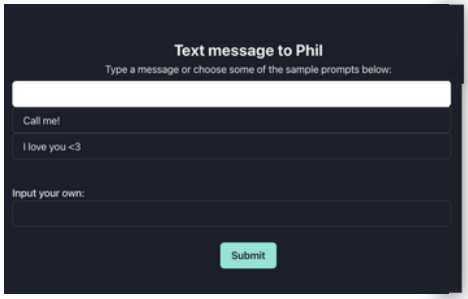


Figure 3.15 - Text Message Screen

Personal Information

Within the MagicMirror Family Sim, users also have a way of:

1. Checking their own mailbox,
2. Checking their own calendar
3. Changing their own settings

This information can be accessed by selecting their own avatar on the right-hand side of the screen.

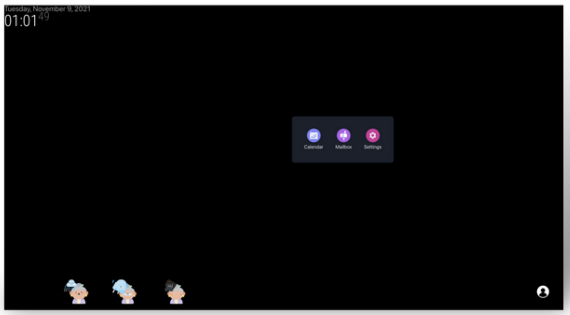


Figure 3.15 - Text Message Screen

Checking your own mailbox

In your own mailbox, you're able to read and listen to all the messages that users send you. Users' avatars appear as your connected users send voice and text messages. TO listen to them you need only click on them.

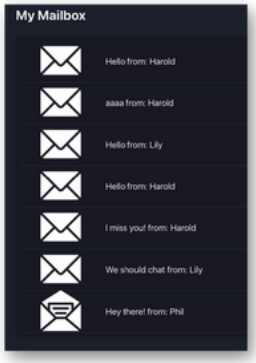


Figure 3.16 - Mailbox: List of Messages

Checking your own calendar

Your own calendar is provided to show you all your calls scheduled with other connected users. This version of the calendar can be used to schedule calls, with the pop-up allowing you to specify who you want to call and when.

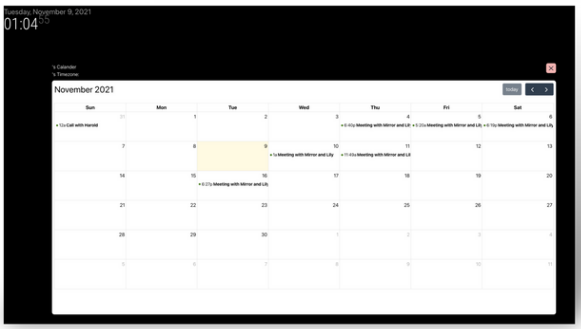


Figure 3.17 - Personal Calendar

Changing your own settings

The settings menu allows you to change your avatar, and update your privacy settings which can change who is able to view your scheduled events etc. You can also update your descriptions for other users, add call prompts and connect other bluetooth devices.

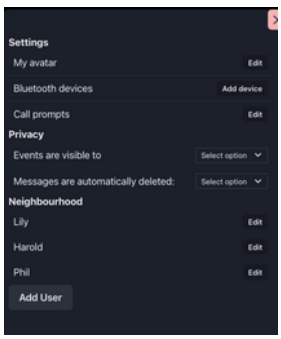


Figure 3.18 - Personal Settings

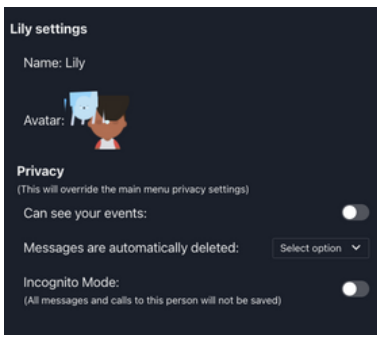


Figure 3.19 - Neighbourhood Users' Settings

4. Hardware/Software Requirements

The requirements needed include:

- Webcam
- Consistent WiFi/Ethernet Connection
- Raspberry Pi 4
- `.env` file for Twillo Video Call

Software

The system is a Linux-based Electron application equipped with a React layer uptop the Electron applet. It utilises the MagicMirror open source project, which allows for custom components to be written and positioned atop the Electron scaffold they provide. By rendering a React component in the lower third - we gain all the benefits of React (such as Components, Stateful refreshes and `npm` package management. The platform takes advantage of the MagicMirror open source project, Chakra UI library, Lottie animation Library, Twillo API service for Video Chat, and the fullcalander.io library for the Calendar component. References to versions can be found in the react-app's `package.json` file.

Our backend system is built on top of AWS using the Amplify backend framework, which allows for quick prototyping of cloud infrastructure. We utilise a DataStore API connected to NoSQL DynamoDB that represents Events, Users, Calendars and Messages. Every User has a 1:1 relationship with the Calendar and a 1:many relationship to Events, and every Calendar is equipped with a 1:many relationship with Events. An indepth summary of what attribute each object contains can be found in the `models/index.d.ts` folder

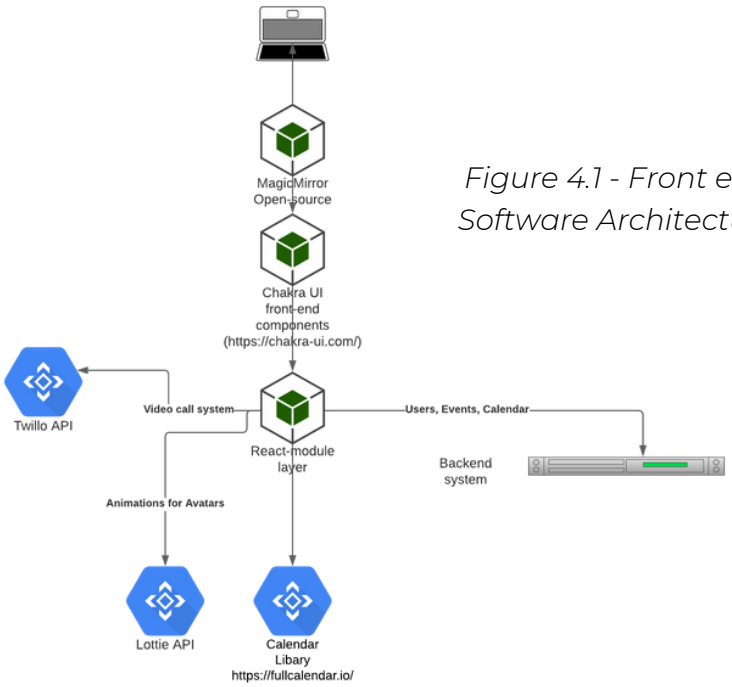


Figure 4.1 - Front end Software Architecture

Hardware

The prototype is a simple computer monitor of any size, with the screen behind a two-way mirror film such that it only displays the non-black elements of the user interface. The frame around the monitor is the user's choice, and the only consideration that has to be made is to leave an area for the webcam to be placed/built-in.

Through the Electron Applet running in a Linux based environment, we can leverage the capabilities of the Raspberry Pi 4 to provide the computation and communication medium for the device. The HDMI cable from the Pi to the full-screen Electron app running in the browser connects this front-end to the user behind the two-way mirror film through the inbuilt computer monitor. Due to computer monitors coming in all sizes now, this system is readily available to fit any size monitor available on the market.

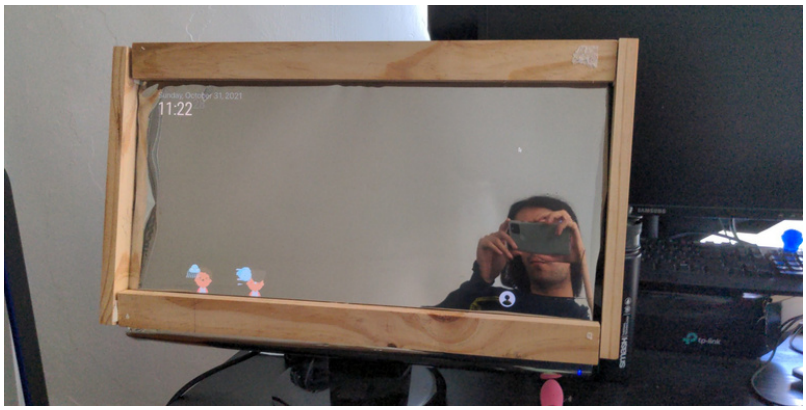


Figure 4.3- Physical implementation

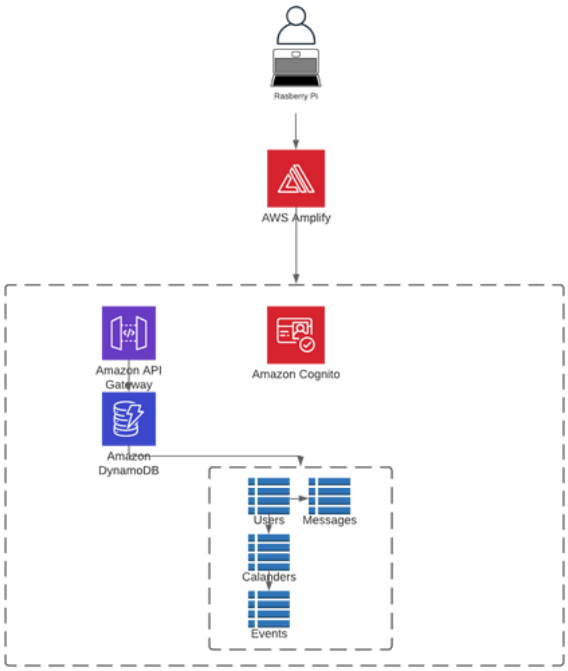


Figure 4.2 - Back-end System Architecture

5. Illustrated Set-up Instructions

Hardware
Setup a Linux based System (Mac or any Linux distro) that can utilise the CLI tools Yarn and NPM. Instructions for CLI setup can be found in the README of the repo.

Attached below is the content for `.env`. This manages the Twillo Video-calling API that allows for local/network-based video chat. The API Key and Secret are found there and need to be stored in an .`env` file. Instructions for this can also be found in the repository.

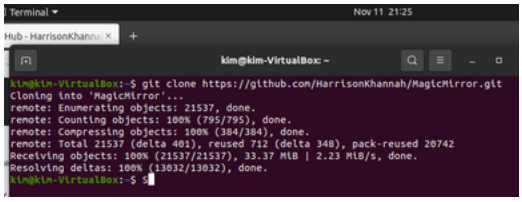
.env content (make this into a .env file!):

TWILIO_API_KEY=SKda5696da450442489a37e4a0ce19fdd8
TWILIO_API
_SECRET=j61ReribPC0YgJU2dZ9NUzeY9v491hjg
TWILIO_CHAT_SERVICE_SID=
TWILIO_TWIML_APP_SID=
TWILIO_ALLOW_INCOMING_CALLS=

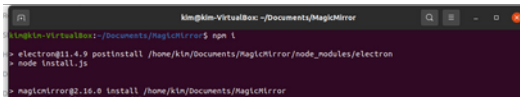
Repository Link: https://github.com/HarrisonKhannah/MagicMirror

Step-by-step Instructions

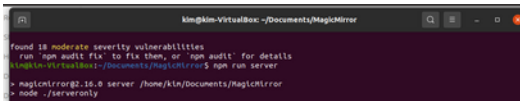
1. Clone the repository into your local files using a git terminal or a GUI repository tool such as Github Desktop.



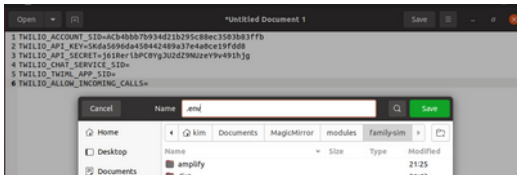
2. Open a terminal at the root level of the repository and run "npm i" in the command line.



3. Next, run "npm run server" at the same level.

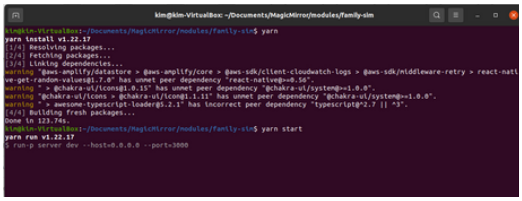


4. Copy the .env content and paste it into a code editor. Name and save the file as ".env" at the modules/family-sim level.



5. Open a new terminal at modules/family-sim level and run "yarn" in the command line.

6. At the same level, run "yarn start". A HTML browser should automatically open with the interactive prototype running after the build has completed in the terminal window.



6. Known Issues

Windows Compatibility

Currently the Mirror utilises the open source project MagicMirror. This open source project is to be used on a Linux-based platform, being that the MagicMirror is not able to be run on a non-Linux platform, such as Windows. We look to make it compatible with Windows in future versions by manufacturing our own forked version of the MagicMirror open-source scaffold.

Local-host vs Network Based Video Chat

The platform does support the use of localised Video Chat on a private network, as seen below. This setup is not the default in the pull-down of the repository (naturally as network IP is different). By instating the local `node` server and frontend in the `React`to not point to your local network, and joining the same Event on two machines on the same network, this function can be tried and tested. However, on initialization of the repo, the video call will not be able to be joinable by other participants on your network. If you choose to enable this, alterations to your `chrome://flags` will have to occur to allow webcam/microphone use on a non-secure connection.

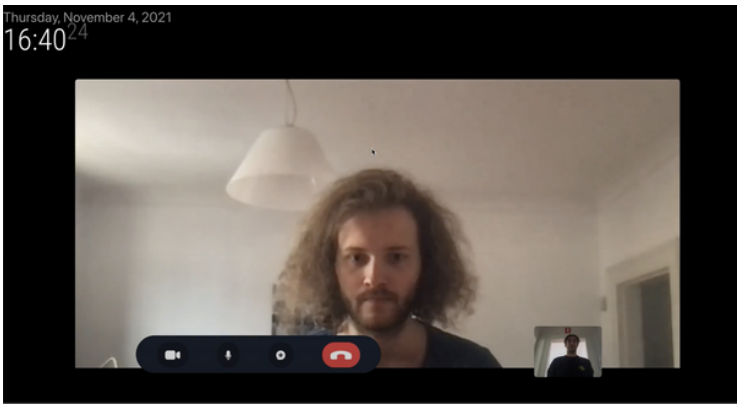


Figure 5.1 - Video Call

Chrome Security Settings

Chrome, by default, requires a secure connection to allow for microphone, video calling and chat. As this is a prototype, to fully test the video calling feature, alterations to your “chrome://flags” have to occur. Please see here(https://medium.com/@Carmichaelize/enabling-the-microphone-camera-in-chrome-for-local-unsecure-origins-9c90c3149339) for a detailed summary on how to add the unsecure local url to allow webcam and microphone usage.

Animation Glitch

The applet runs using the Bodymovin plugin which is compatible with exported media from After Effects, however appears to glitch in actual implementation.

Figure 5.2 - Avatar Animations

7. Future Work & Versions

There are a few future concepts that are to be included into the future iterations of the report. These extra concepts provide the users with more activities for further enjoyment, and also provide accessibility aspects for further ease of use for both able-bodied and the impaired.

Games

As a team, we've tested the idea of being able to play scrabble and chess with other users whilst on call which brought back positive responses. These games were decided as they can be played asynchronously, and don't need intensive live interaction to keep the immersion present. This and further other games will be included in the future that can improve user enjoyment.

Motion & Voice/Touch Technology

Motion technology will help users interact with the mirror. As a wide range of age demographics interact with this device, motion technology in the form of gesture and voice interactions being registered by the camera will help them from having to reach and strain themselves. Touch technology will also help the young demographics, as the touch model would make interaction even easier. By implementing models of interactions that are more familiar and easier for our users, we can ensure that maximum usability is achieved.

Bluetooth with Cochlear Implant

Bluetooth connectivity was incorporated into the design primarily to connect to sound peripherals that can allow users to hear other users better. Currently it connects effectively to sound devices such as speakers however in future iterations, we plan to have support for connecting to cochlear implants via bluetooth to help those who have hearing impairments and can benefit from having direct connections.

MagicMirror Mirror/Companion Application

In future, a potential need for a companion application has been raised due to the accessibility it provides for those who may struggle with mobility in accessing the device. By providing a mirror application, particular elderly will not have to strain themselves with relatively large movements compared to that performed on a phone. Alternatively, a companion application will provide all users with the ability to change settings and avatars whilst on the go for added convenience.

8. References

Alce, G., Wallergård, M., & Hermodsson, K. (2015, June 10). Wozard: A wizard of oz method for wearable augmented reality interaction-a pilot study. Retrieved September 11, 2021, from https://www.hindawi.com/journals/ahci/2015/271231/

Bland, D. (2020). What is an Empathy Map. Retrieved 29 September 2021, from https://www.accenture.com/us-en/blogs/software-engineering-blog/what-is-an-empathy-map

Bono, E. . (1988). Letters to thinkers: Further thoughts on lateral thinking. London: Penguin Books.

Chang, K. (2015). Decisions in Engineering Design. Retrieved November 11, 2021 from https://www.sciencedirect.com/topics/engineering/decision-matrix

Chung, E. (2020). Generate Crazy Ideas With This Design Sprint Method. Retrieved November 11, 2021 from https://uxplanet.org/generate-crazy-ideas-with-this-design-sprint-method-c6a36a16c3d5

Davies, M. (2010). Concept mapping, mind mapping and argument mapping: what are the differences and do they matter?. Higher Education, 62(3), 279-301. doi: 10.1007/s10734-010-9387-6

Design Council. (2019). What is the framework for innovation? Design Council's evolved Double Diamond. Retrieved 11 November 2021 from https://www.designcouncil.org.uk/news-opinion/what-framework-innovation-design-councils-evolved-double-diamond

Gilbert, K. (2018, October 10). The co-design Workshop: The FACILITATOR'S Pocket Guide. Retrieved September 11, 2021, from https://connection.domain7.com/the-co-design-workshop-the-pocket-facilitators-guide-e36a6c9e08d4

Holzmann, C., & Hutflesz, P. (2014). Multivariate Testing of Native Mobile Applications. Proceedings Of The 12Th International Conference On Advances In Mobile Computing And Multimedia, 85–94. doi: http://dx.doi.org/10.1145/2684103.2684119

Kohavi, R., & Longbotham, R. (2016). Online Controlled Experiments and A/B Testing. Encyclopedia of Machine Learning and Data Mining. doi:10.1007/978-1-4899-7502-7 891-1

Krause, R. (2018). Storyboards Help Visualise UX Ideas. Retrieved from November 11, 2021 from https://www.nngroup.com/articles/storyboards-visualize-ideas/

Laubheimer, P. (2016). Wireflows: A UX Deliverable for Workflows and Apps. Retrieved 7 October 2021, from https://www.nngroup.com/articles/wireflows/

Moran, K. (2018, January 21). Writing Tasks for Quantitative and Qualitative Usability Studies. Retrieved September 14, 2021, from Nielsen Norman Group website: https://www.nngroup.com/articles/test-tasks-quant-qualitative/

Nicholson Jr., N. R. (2009). Social isolation in older adults: An evolutionary concept analysis. *Journal of Advanced Nursing, 65*(6), 1342-1352. doi:10.1111/j.1365-2648.2008.04959.x

Pedell, S., Beh, J., Mozuna, K., & Duong, S. (2013). Engaging older adults in activity group settings playing games on touch tablets. *Proceedings of the 25th Australian Computer-Human Interaction Conference on Augmentation, Application, Innovation, Collaboration - OzCHI '13*. doi:10.1145/2541016.2541090

Rosala, M. (2021). Using "How Might We" Questions to Ideate on the Right Problems. Retrieved November 11, 2021, from https://www.nngroup.com/articles/how-might-we-questions/

Salazar, K. (2020). Contextual Inquiry. Retrieved 7 October 2021, from https://www.nngroup.com/articles/contextual-inquiry/

Situ, A. (2019). Wild Cards: A Tool for Ideation and Brainstorming. Retrieved from November 11, 2021 from https://blog.prototypr.io/ideation-cards-wild-cards-b3ebf5a9d687

Tomitsch, M., Borthwick, M., Ahmadpour, N., Cooper, C., Frawley, J., . . . Wrigley, C. (2018). Design. think. make. break. repeat: A handbook of methods. Amsterdam: BIS.