

CREATING SOFTWARE THAT WORKS FOR EVERYONE

From monitoring our health to finding a parking spot, there are now apps to help with every task in life. However, as we become ever-more reliant on software, ensuring it is accessible for all of society is a growing necessity. At **Monash University** in Australia, the **Human-Centric Software Engineering (HumaniSE) Lab** is developing software that can be used by everyone.



HumaniSE Lab

Human-Centric Software Engineering (HumaniSE) Lab, Monash University, Australia
www.monash.edu/it/humanise-lab

Field of research

Software engineering

Research project

Developing accessible software for diverse end users

Funder

Australian Research Council

TALK LIKE A ...

SOFTWARE ENGINEER

ACCESSIBLE — can be obtained easily and used by everyone

APP — a computer program used to perform a specific function (e.g., finding and paying for a parking space)

END USER — the person who uses an app or software

HUMAN-CENTRIC — with human needs and values as key considerations

SOFTWARE — the set of programs and other information that enables a computer or app to function

SOFTWARE DEVELOPER or SOFTWARE ENGINEER — someone who designs and develops software

USER INTERFACE — the place where an end user interacts with a computer (e.g., a webpage or app display)

“Every human is different,” says Professor John Grundy. “They have different ages, languages, cultural backgrounds, personalities, and physical and mental characteristics. Software should take these differences into account.” However, most software does not. Developers commonly design software with a one-size-fits-all approach, based on their own assumptions of what end users want from the software. This results in software that is not accessible to everyone, leaving users confused or frustrated when trying to interact with the software, or excluded from it entirely.

To address this problem, John founded the Human-Centric Software Engineering (HumaniSE) Lab at Monash University. Through a variety of creative approaches, John’s team is developing inclusive software that considers the unique needs and characteristics of today’s diverse populations.

Software challenges

The range of different needs and preferences exhibited by people around the world means that ensuring software accounts for all these differences is no easy task. A lot of existing software is not accessible to certain demographics, which raises issues for equity and inclusion.

For example, elderly people may struggle to use certain technologies because they are unfamiliar with them, do not understand how to navigate software, cannot read small text or do not have the dexterity to control a touchscreen. Dyslexic users may struggle to read text, colour-blind users may struggle to understand images and graphics, and users with visual impairments may be unable to access any visual information.

“People are not all the same, so software cannot assume one kind of end user,” says Dr Hourieh

Khalajzadeh, a member of the HumaniSE Lab.

“Software should take these differences into account.” For example, inclusive software should support different colour schemes, font styles, text and button sizes, interaction styles (e.g., written text or spoken voice) and languages. Most importantly, to be accessible, software should allow the user to adapt the user interface according to their personal preferences.

These issues highlight the human element of software, both in production and consumption. As software is developed by people who are very well-versed in how software functions, they may not appreciate the challenges that others face when interacting with software.

“Software developers face numerous challenges when trying to engineer more human-centric software,” explains Dr Anuradha Madugalla, another member of the team. “It can be hard to interact with the range



© mangpor2004/stock.adobe.com

of target end users, meaning developers struggle to understand the issues users face and are unable to collect feedback on the usability of their software.” The HumaniSE Lab hopes to address these challenges by creating personas of diverse end users to ensure the software it develops meets the needs of the target audience.

What are personas?

“Personas are fictional representations of software users,” explains Hourieh. “They are used to help developers think about catering to different demographics with different user goals and desired features.” While not as informative as testing software on real people, personas are a practical solution when real end users are unavailable to give feedback.

“Without consideration of diverse personas, there is a danger that the needs of some end users are missed during software development,” says Anuradha. In the HumaniSE Lab, the team creates personas and associated ‘user stories’ that identify the different requirements end users may need from the software.

Developing a parking app

The HumaniSE team has put its methods to use by developing an accessible parking app to help drivers find and pay for parking. After reading

“**PEOPLE ARE NOT ALL THE SAME, SO SOFTWARE CANNOT ASSUME ONE KIND OF END USER. SOFTWARE SHOULD TAKE THESE DIFFERENCES INTO ACCOUNT.**”

reviews of current parking apps, which informed the team of the issues with current apps, a set of representative personas and user stories were created that identified the requirements different people have of a parking app.

Examples of specific personas and user stories include: a student who wants a quick and easy way to book a parking spot and whose priority is having a secure payment method; a young adult with English as their

second language who requires multilingual support and wants a clean and simple user interface; an aging user with mobility challenges who wants information to be provided in a large font size and for the app to find an accessible parking spot that reduces the walking distance to their final destination; and a colour-blind user who owns multiple cars who needs an accessible colour scheme and who wants to easily switch between different car registrations when paying for parking.

“Using these requirements, we designed a prototype parking app that would meet all the needs of all target user groups, represented by the different personas,” explains John. “We then tested our prototype app against each persona to see if the personas’ goals were fulfilled and their frustrations prevented.” This involved a ‘cognitive walk-through’ of the app, with different team members taking the role of each persona and assessing whether the app met their requirements. Ideally, the team would have evaluated the app by working with people from the different target user groups and collecting their feedback, but the COVID-19 pandemic prevented this during the app development stage.

While personas are a powerful tool, they cannot necessarily capture all end user requirements given the massive diversity of people’s needs and preferences. “Using personas to assess the app can only give some indications of feature suitability, as different people experience different challenges in different ways,” explains Anuradha.

Through this process, the team developed a human-centric parking app that offers multiple ways to find and book a parking space, addresses concerns around data privacy and payment, and has an adaptable user interface allowing the user to choose the language, colour scheme, font and interface complexity.

The importance of inclusive health apps

Increasingly, health services and people with health conditions are being supported by apps. Technology provides a powerful tool for healthcare. It can supply specific information about illnesses, give reminders



©Peakstock/stock.adobe.com



© pressmaster/stock.adobe.com

to take medication, capture and monitor health-related measurements such as heart rate, alert health professionals to emergencies and provide support to maintain a healthy lifestyle. Nevertheless, for this potential to be fully realised, accessibility of health apps needs to be a top priority.

“Different people experience illnesses and chronic conditions in different ways, so health apps must be adaptable to the individual’s needs,” says Hourieh. “Health conditions also change over time in different ways, and apps must adjust for this.” Given that the management of illnesses is critical for people’s well-being, ensuring that health apps appropriately cater to everyone is a key concern. A one-size-fits-all approach to healthcare apps can even be dangerous as everyone has their own unique healthcare needs.

The HumaniSE Lab has undertaken several projects related to the accessibility of health apps. After reviewing existing guidelines for health app development, the team identified gaps in current protocols that failed to address the needs of diverse end users. “We then worked with a number of health app development teams to use human-centric software development practices,” says Anuradha. “These included participatory design (where target app users are involved in the app design process), using human-centric personas, creating adaptive user interfaces, and establishing a review process to detect, analyse and fix any issues.”

“ DIFFERENT PEOPLE EXPERIENCE ILLNESSES AND CHRONIC CONDITIONS IN DIFFERENT WAYS, SO HEALTH APPS MUST BE ADAPTABLE TO THE INDIVIDUAL’S NEEDS. ”

The outcomes of these projects have led to the development of guidelines and support tools for software developers to use when designing future health apps. These help developers to identify human-centric issues and develop health apps that address the needs of diverse end users. Software developers around the world are therefore benefitting from the work of the HumaniSE Lab, as they now have the means to ensure their health apps are suitable for all users. As a result, the people who use apps to help manage their health are also benefitting from technology that is inclusive, accessible and adaptable.

What next?

Through the HumaniSE Lab’s research, it has become increasingly apparent to the team that a more inclusive approach to software design is needed. Including target end users in the design phase, rather than just collecting their feedback once the software has been

“ WE ARE AIMING TO SET UP A ‘LIVING LAB’ TO TAKE A MORE CO-CREATIONAL APPROACH TO HUMAN-CENTRIC SOFTWARE DEVELOPMENT. SOFTWARE DEVELOPERS AND TARGET END USERS WOULD CLOSELY COLLABORATE TO DESIGN, TEST AND REFINE SOFTWARE REQUIREMENTS. ”

developed, leads to the creation of software that considers user accessibility at every stage, rather than relying on tweaks at the end of the process that might not satisfactorily address usability challenges.

The team has a bold vision for bringing this approach into reality. “We are aiming to set up a ‘Living Lab’ to take a more co-creational approach to human-centric software development,” says John. “Software developers and target end users would closely collaborate to design, test and refine software requirements.” A key concept of the Living Lab is that all participants, both developers and users of software, are fully involved and treated as equals throughout the software development process, ensuring the needs and opinions of the users and developers are always considered. “We want to use the Living Lab as a place to create better software for everyone.”

Pathway from school to software engineering

- “At school, take any subjects you enjoy!” advises John, as most universities will not expect you to have a software engineering background before beginning your degree. However, if you are interested in a career in software engineering, it would be useful to study mathematics, computer studies/ICT and physics at school. These will provide useful knowledge, develop your problem-solving skills and may be requirements for some university degrees.
- Hourieh advises learning programming as soon as possible by exploring the range of free courses and tutorials available online. “Block-based programming (which is visual-based rather than code-based) can give students a good overview of how programming works,” she explains. “Scratch (www.scratch.mit.edu) and Code (www.code.org) are great tools to learn the fundamentals of computer science by creating your own games and animations.” Once you understand the concepts of programming, you can then learn coding languages, such as Python.
- At university, many degree options could lead to a career in software engineering. These include degrees in software engineering, computer science, information technology, data science and information systems.
- “Alongside your computing degree, I would recommend taking a course in psychology, as this will help you become a software engineer who understands your end users,” advises Anuradha. John agrees that combining studies in computing with science, social science or business courses will not only give you a broader education but will allow you to apply your software engineering skills in a wider range of applications (e.g., finance, manufacturing, agriculture, etc.).
- If you want to train and learn skills while working, look for organisations that offer software-related apprenticeships.

ABOUT SOFTWARE ENGINEERING

Software engineering, or software development, is the branch of computer science that involves designing and developing software – the instructions and programs that enable computers to function. The field of software engineering is growing rapidly as computers are becoming integrated into all aspects of our lives.

The joys of software engineering

The HumaniSE Lab members all enjoy their work as software engineers due to the satisfaction, challenges and diversity of the job. “I like what I am doing as I believe I am making life better for humanity,” says Anuradha. Hourieh likes the fact that software engineering is “a challenging job that involves continuously learning new skills and improving your capacities for problem-solving and creativity”.

“I like the range of problems I get to work on,” says John. “I have worked on software development for health, manufacturing, local government, education and games. You can get involved in lots of different disciplines, meet a wide variety of people and build software solutions that make a big difference in people’s lives.”

Why should you consider a career in software engineering?

If you have software engineering skills, you can work anywhere in the world, in any sector. “Software engineering pays well and has great long-term career prospects,” says John. “You can work on globally important problems, such as addressing climate change through building software for energy-efficient ‘smart’ buildings.”

Hourieh highlights how the diversity of options

in software engineering gives you scope to work in your preferred role. “Is it the design part that most interests you?” she asks. “Or do you enjoy coding? Or would you like a role in project management?” There are software engineering positions for all of these. “Software engineering gives you the opportunity to make your ideas a reality with only the effort of your mind and fingers,” says Anuradha.

With the global rise in automation and artificial intelligence, and an ever-increasing amount of data being collected about end users, the next generation of software engineers will have to ensure all technologies are designed and used responsibly and ethically. There will be challenges surrounding privacy and security to overcome, along with the issue of ensuring future software is accessible for everyone. Where could a career in software engineering take you?

“SOFTWARE ENGINEERING PAYS WELL AND HAS GREAT LONG-TERM CAREER PROSPECTS. YOU CAN WORK ON GLOBALLY IMPORTANT PROBLEMS, SUCH AS ADDRESSING CLIMATE CHANGE THROUGH BUILDING SOFTWARE FOR ENERGY-EFFICIENT ‘SMART’ BUILDINGS.”

PROFESSOR JOHN GRUNDY

Explore careers in software engineering

- As technology is now so ingrained in our lives, software engineers are needed to develop software in all fields. Depending on your interests, you could create the next social media app, design software to reduce energy use or program a new virtual assistant.
- Hays provides information about the type of work software engineers do, the skills you will need and the salary you can expect: www.hays.com.au/it/software-engineer-jobs
- Participate in an internship or industry placement to gain real-world experience of software engineering. You will learn how software teams work, how businesses function and about the diverse range of end user needs.
- To connect with software engineers and learn about careers in the field, John recommends joining a professional society as a student member, such as Engineers Australia (www.engineersaustralia.org.au) or the Australian Computer Society (www.acs.org.au). International societies include the Institute of Electrical and Electronics Engineers (www.ieee.org) and the Association for Computing Machinery (www.acm.org).
- Anuradha suggests that girls interested in software engineering should explore Girls Who Code (www.girlswhocode.com) and Women in Engineering (www.wie.ieee.org), organisations helping women engineers to reach their potential.

Meet the HumaniSE team



Professor John Grundy

Professor of Software Engineering, Monash University, Australia

I was first exposed to computers in my mid-teens. At university, I planned to major in chemistry or progress to medical school. I enjoyed my first-year computer science courses so much that I switched paths as I decided building software would be more satisfying and enjoyable. I have since worked on many health systems, so have still had the opportunity to contribute to healthcare.

I enjoyed playing cricket and football when I was young. I even created animated cricket and football computer games in my teens! I have also always been fascinated by military and ancient history, and I would love to combine these interests with software engineering. Perhaps I will develop software to analyse historical artefacts one day.

As an undergraduate, I enjoyed developing computer games, robot programs and software to manage information. Completing an industrial placement at a software company, working on financial systems, showed me the huge flexibility and good pay offered by a career in software engineering.

Software engineering is more people-oriented than most people imagine. I enjoy working with other software engineers and the end users of our software. I have been fortunate to work with a wide range of people over the years on all sorts of interesting software development projects.

I face a number of challenges using software myself, and I have many family members and friends who also have challenges using software. These include colour-blindness, hearing issues, cognitive challenges, neuroatypical problem-solving styles and different languages. This has led to me being passionate about ensuring software is human-centric.

I founded the HumaniSE Lab after winning an Australian Laureate Fellowship in 2019. This provided funding for several researchers and PhD

students to work with me on developing better ways to engineer human-centric software. The lab is a focal point for this sort of research and development. Establishing the HumaniSE Lab has been a great highlight of my career.

In my free time, I enjoy running, watching cricket and football, reading about ancient Egypt, and listening to audiobooks and 80s music. I also love spending time with my wife and children.

John's top tips

1. Do what you enjoy doing – you can apply software engineering to almost any other field!
2. Find people who accept and respect you for who you are, and do the same for them.
3. Keep a good work-life balance. Overworking leads to mistakes, which could be catastrophic (you want the person who designed and tested the software in your car to have been concentrating while they did it!). Besides, time with family and friends and pursuing hobbies is hugely valuable.

“I FOUNDED THE HUMANISE LAB AFTER WINNING AN AUSTRALIAN LAUREATE FELLOWSHIP IN 2019. THIS PROVIDED FUNDING FOR SEVERAL RESEARCHERS AND PHD STUDENTS TO WORK WITH ME ON DEVELOPING BETTER WAYS TO ENGINEER HUMAN-CENTRIC SOFTWARE.”

PROFESSOR JOHN GRUNDY



Dr Hourieh Khalajzadeh

Lecturer in Software Engineering, Deakin University, Australia

I wasn't especially interested in computers as a young kid. Initially, I wanted to go to medical school because everyone around me said I should. Then, when I discovered I loved maths, I wanted to be a mathematician. When I started learning programming, I really enjoyed it, so I decided to study software engineering at university.

I was the first girl in my family to become an engineer. I'm pleased to say I'm not the last! It was my interest in maths and programming that inspired me to become an engineer, and software engineering was the branch that was closest to my heart. At university, I met incredible female academics who motivated me to continue with postgraduate studies.

Completing a master's, PhD and postdoc all motivated me to pursue software engineering research. I am fortunate that John supervised me during my PhD as he is a top researcher in the

field. When he established the HumaniSE Lab, I was one of the lucky researchers who was selected to join the team.

Being a researcher gives you the freedom to work on the topics you most enjoy, collaborate with the people you enjoy working with, find the gaps that others haven't been able to explore, and solve the problems that can improve people's lives. I became interested in computer science research as I enjoy exploring the untapped areas in different domains where software can be a life changer.

We design software to help people. This means we must ensure it is inclusive of everyone, no matter what they do, where they are located, how old they are, what language they speak, and what preferences and characteristics they have. If software is designed based on the developer's own assumptions of end users' needs, then the software may be unusable or even dangerous. For instance, if a health application recommends a solution that is not appropriate for a specific user, it could put their life at risk. This is why I am passionate about ensuring software is human-centric.

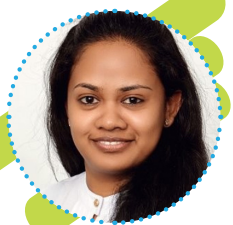
I have developed many tools that have been used in software for different applications, including

health, transportation and banking. I aim to make not only software, but also software development, more accessible for everyone, even those without a technical background or without access to the latest infrastructure.

When I'm not working, I enjoy spending time with my family and friends. We do lots of outdoor activities such as hiking and cycling. I also enjoy travelling to new places and experiencing new things – most recently, skydiving and scuba diving!

Hourieh's top tips

1. Do what you most enjoy doing.
2. Explore different career pathways to find the one most suitable for you. Talk to people from different professions, take a range of courses and attend information sessions to find what interests you.
3. If you are interested in software engineering, follow the latest technology news to stay aware of what's happening around us and what changes software engineers can make.



Dr Anuradha Madugalla

Research Fellow in Software Engineering, Monash University, Australia

I used a computer for the first time when I was eight, to play games. As I grew older, though I liked computers, my interests were broader and covered all technology-related things. This led me to study IT as my university degree, which is when I started to truly fall in love with software.

I always wanted to be the first to set up any new technological device in the house and play with it. My mother saw my love of technology and encouraged me to consider a future in IT. It was her belief and my personal interest in technology that led me to software engineering.

Growing up, there were not many girls doing software engineering around me. I love a challenge, so this motivated me to pursue this career path and to perform well in it. Many great women in IT, such as Anita Borg and Sheryl Sandberg, have inspired

me to continue working in software engineering.

My research has always focused on developing software to make lives better. This has included software that can reconstruct faces from skulls, to help forensic scientists identify victims, and software to help people with visual impairments understand graphics. I first joined the HumaniSE Lab to develop tools and techniques for aged care, and now I help to develop human-centric software for all sectors of society.

During my PhD, I worked closely with many blind people as I helped develop software for people with vision impairments. I accompanied them as they visited our lab and visited their homes to collect their feedback on our software. I experienced how challenging life can be for blind people and was fascinated by how well they overcame these challenges. I was impressed with how blind people use software such as screen readers but realised how much they are missing when software is not accessible to their needs. This inspired me to think more about how to make software more user-focused and helpful, not only for those with vision impairments, but for all end users with different challenges depending on their unique characteristics.

I am proud of many of my achievements in my studies and career. These include winning a gold medal for the best undergraduate research, winning the Faculty of IT's Three Minute Thesis (3MT) competition where I presented my PhD research in just three minutes, and winning the Google Women Techmakers award.

I love reading in my free time. I enjoy all types of fiction, from magical fantasies to thrillers, historical fiction to classic literature. I also love travelling to see the world and spending time with my family.

Anuradha's top tips

1. If software engineering and computers interest you, start learning programming and coding as soon as you can.
2. Work hard at school to achieve the best grades you can, especially in maths.
3. Enjoy life as best you can, as it will go by without you realising!