Original Paper

Leveraging Generative AI Tools to Support the Development of Digital Solutions in Health Care Research: Case Study

Danissa V Rodriguez[1], PhD; Katharine Lawrence[1,2], MPH, MD; Javier Gonzalez[2], BS; Beatrix Brandfield-Harvey[1], BA; Lynn Xu[1], MPH; Sumaiya Tasneem[1], MPH; Defne L Levine[1], MPH; Devin Mann[1,2], MS, MD

[1]Department of Population Health, New York University Grossman School of Medicine, New York, NY, United States

[2]Medical Center Information Technology, Department of Health Informatics, New York University Langone Health, New York, NY, United States

Corresponding Author:
Danissa V Rodriguez, PhD
Department of Population Health
New York University Grossman School of Medicine
227 East 30th Street
6th Floor
New York, NY, 10016
United States
Phone: 1 646 501 2684
Email: danissa.rodriguez@nyulangone.org

Abstract

Background: Generative artificial intelligence has the potential to revolutionize health technology product development by improving coding quality, efficiency, documentation, quality assessment and review, and troubleshooting.

Objective: This paper explores the application of a commercially available generative artificial intelligence tool (ChatGPT) to the development of a digital health behavior change intervention designed to support patient engagement in a commercial digital diabetes prevention program.

Methods: We examined the capacity, advantages, and limitations of ChatGPT to support digital product idea conceptualization, intervention content development, and the software engineering process, including software requirement generation, software design, and code production. In total, 11 evaluators, each with at least 10 years of experience in fields of study ranging from medicine and implementation science to computer science, participated in the output review process (ChatGPT vs human-generated output). All had familiarity or prior exposure to the original personalized automatic messaging system intervention. The evaluators rated the ChatGPT-produced outputs in terms of understandability, usability, novelty, relevance, completeness, and efficiency.

Results: Most metrics received positive scores. We identified that ChatGPT can (1) support developers to achieve high-quality products faster and (2) facilitate nontechnical communication and system understanding between technical and nontechnical team members around the development goal of rapid and easy-to-build computational solutions for medical technologies.

Conclusions: ChatGPT can serve as a usable facilitator for researchers engaging in the software development life cycle, from product conceptualization to feature identification and user story development to code generation.

Trial Registration: ClinicalTrials.gov NCT04049500; https://clinicaltrials.gov/ct2/show/NCT04049500

(JMIR Hum Factors 2024;11:e52885) doi: 10.2196/52885

KEYWORDS

digital health; GenAI; generative; artificial intelligence; ChatGPT; software engineering; mHealth; mobile health; app; apps; application; applications; diabetes; diabetic; diabetes prevention; digital prescription; software; engagement; behaviour change; behavior change; developer; developers; LLM; LLMs; language model; language models; NLP; natural language processing

Introduction

Health care has undergone a digital transformation, resulting in a growing reliance on software engineering for medical use cases, including health care research. However, little guidance exists for health researchers on how to effectively develop digital health interventions [1]; in particular, software development challenges that include expertise gaps in coding, custom development needs, high costs, and time constraints result in

XSL•FO
RenderX

multilevel barriers to designing and deploying a usable, scalable, and sustainable digital health product [1].

Generative artificial intelligence (GenAI) technologies such as ChatGPT can potentially support researchers in health technology endeavors by providing foundational frameworks and processes for the software development life cycle [2]. These systems can help reduce time and enhance precision for technology-based research projects by supporting both nonprogrammers and experienced programmers in code development, troubleshooting, and cleaning [2]. Moreover, the ability to use GenAI to generate content from different perspectives (expert or nonexpert) can facilitate and improve communication between technical and nontechnical team members of multidisciplinary teams. For example, a nontechnical team member can write their ideas in natural text and then use GenAI to request assistance in creating discussion points to communicate to a technical team audience. GenAI tools may also help health technology researchers refine research questions, identify appropriate theoretical frameworks and models, and leverage popular implementation strategies such as design thinking to build effective, theory-grounded, and evidence-based digital health interventions. ChatGPT (OpenAI, Microsoft Corporation) has already demonstrated feasibility as a support tool for clinical decision support development in health care [3], and more broadly as a coding copilot in programming and engineering [4,5].

This study explores the use of ChatGPT to recreate a personalized automatic messaging system (PAMS), which was developed as part of a digital health research initiative to support patient engagement with a commercial digital diabetes prevention program (dDPP). We examine the capacity, advantages, and limitations of ChatGPT to support product ideation and conceptualization, intervention content development, and the software engineering process including software requirement generation, software design, and code production. This paper provides insights to support the GenAI-assisted development of computational tools that are usable, reliable, extensible, and in line with the standards of modern coding practices. The framework includes prompts for both the intervention conceptualization as well as the main phases of the software development process.

Methods

Settings and Intervention Development Context

In previous work [6], we described the development of PAMS, a novel integrated multicomponent communications platform, to promote patient-provider communication and patient engagement in a commercial dDPP (Noom; Noom, Inc). The PAMS intervention included early prototyping and user testing, a technical development phase, and a randomized controlled trial. The core content and user experience features of PAMS were identified, prototyped, and evaluated using the well-established design thinking "discover, define, design, and test" approach to iteratively gather information, define, design, and refine the engagement intervention [7]. Stakeholders included: patients with prediabetes and their support network (eg, caregivers and partners), primary care providers, health technologists, programmers and computer scientists, behavioral change theorists and subject matter experts, the research administrative team, and dDPP product developers and coaches. The main components of this PAMS intervention include (1) a theory-driven behavior change messaging library, (2) a personalized automated message system delivery platform (SMS text messaging–based), and (3) EHR-integrated data visualizations. The PAMS messaging library uses an integrated framework that combines established theoretical models for behavior change with human-centered design strategies to maximize the evidence-based conditions for behavior change and the user acceptance and use of a digital health product. The technical development of PAMS followed an agile software development approach based on incremental 2-week sprint cycles consisting of requirement planning, design, development, and testing of a specific set of functional features. In this paper, we will recreate this development process using GenAI (ChatGPT).

ChatGPT-PAMS Experiment Design

To evaluate the effectiveness of using GenAI to support the development of digital tools in medical settings, our experiment is based on recreating PAMS using GenAI (ChatGPT) and evaluating human-generated vs ChatGPT-generated documentation. To accurately capture the ideation and development process, our multidisciplinary team reviewed all documentation and processes used in the early stages of PAMS conceptualization, including supporting theoretical models, content and features, and technical development. We then recreated these processes via a series of prompts for ChatGPT-4 to assist with the generation of theory, content, user stories, requirement documents, design diagrams, and the code for a subset of the requirements. Outputs from ChatGPT were reviewed and compared to human-generated documentation by 11 evaluating team members. Evaluators consisted of clinicians, behavioral scientists, programmers, and research staff working in digital health and technology for behavior change research. Collectively, they represent more than 50 years of clinical, research, design, and computer science experience. The evaluators independently rated the quality of various aspects of information provided by ChatGPT on a Likert scale, where higher ratings indicated greater quality of information (*1: very poor; 2: poor; 3: acceptable; 4: good; 5: very good; N/A: not applicable*). Aspects of evaluation included: understandability (Does this output make sense given the context of the study and prompts?), novelty (Were new ideas generated?) [3], usability (Does this create a usable output?), relevance (Does this create a useful output?), efficiency (Would having these outputs have saved time?), and potential for bias (What unintended consequences might arise from these outputs?) [6]. Evaluators were also asked to give an overall score on the quality of the ChatGPT output (Overall, how good would you say this output is?). Post review, a group debrief was conducted, using a semistructured interview guide to facilitate discussion regarding perceptions of outputs and rationale for ratings.

Ethical Considerations

Ethical considerations helped guide the initial development of research methods and reduce potential risks for participants in

XSL•FO
RenderX

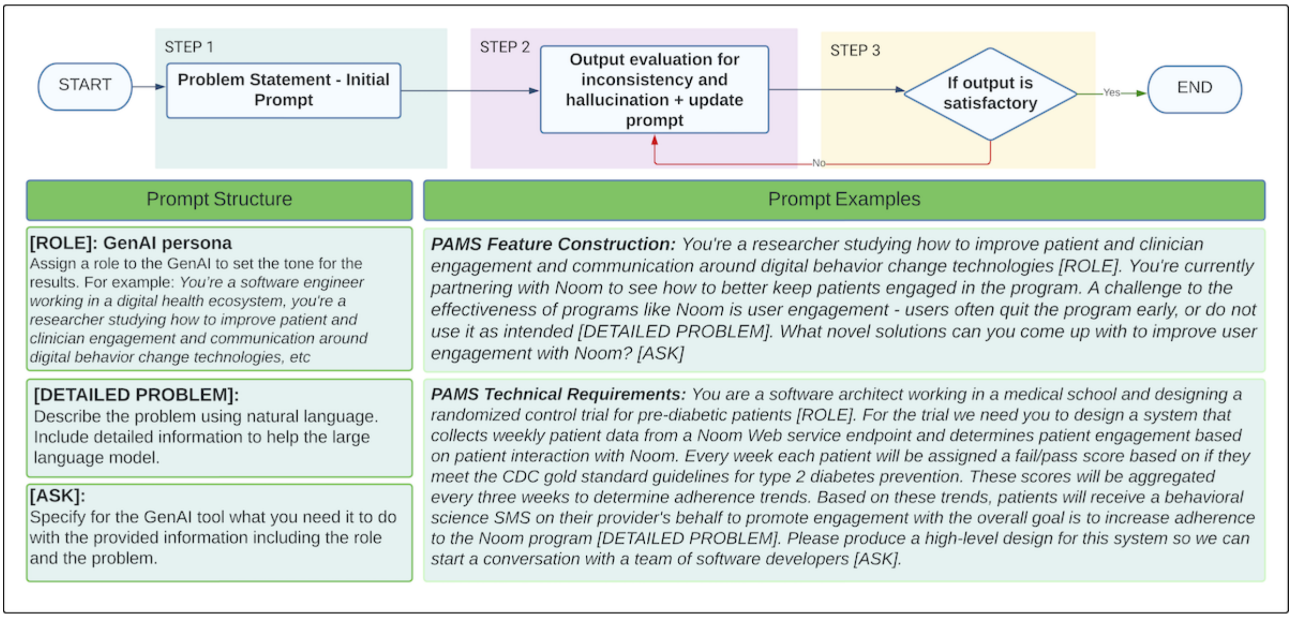
the original study implementation with the PAMS intervention [7]. Recreating the technical development of a system previously built as part of the dDPP randomized controlled trial (NCT04049500) has not introduced any new risks to patients. Patients were not involved in this research examining the use of GenAI in the development of digital health care solutions. No patient data was used in the prompt generation phase.

Regarding ethical considerations for the methods used in this paper, as an attempt to mitigate evaluator biases, we worked with a diverse team of evaluators who were aware of the initial study but were not necessarily involved in the technical development. Additionally, we understand the limitations and concerns of the use of ChatGPT including possible hallucinations and incorrect answers. Thus, we emphasize the need for human expertise to identify correct and incorrect outputs and have flagged this as a study consideration. When developing the GenAI-based solution, we used the same considerations for data security, patient usability, accessibility, and data privacy used in the original human-developed solution.

Prompt Generation Framework

Prompt engineering focuses on the skill of designing and creating effective prompts that guide ChatGPT to produce the best possible output for your task. We followed existing literature [8-11] combined with our expertise and experimentation to provide a framework that yields the best result when developing a digital solution like PAMS (Figure 1).

Figure 1. ChatGPT prompt structure and prompt examples. CDC: Centers for Disease Control and Prevention; GenAI: generative artificial intelligence; PAMS: personalized automatic messaging system.


START
STEP 1
Problem Statement - Initial Prompt
STEP 2
Output evaluation for inconsistency and hallucination + update prompt
STEP 3
If output is satisfactory
Yes
No
END
Prompt Structure
Prompt Examples

[ROLE]: GenAI persona
Assign a role to the GenAI to set the tone for the results. For example: *You're a software engineer working in a digital health ecosystem, you're a researcher studying how to improve patient and clinician engagement and communication around digital behavior change technologies, etc*

[DETAILED PROBLEM]:
Describe the problem using natural language. Include detailed information to help the large language model.

[ASK]:
Specify for the GenAI tool what you need it to do with the provided information including the role and the problem.

PAMS Feature Construction: *You're a researcher studying how to improve patient and clinician engagement and communication around digital behavior change technologies [ROLE]. You're currently partnering with Noom to see how to better keep patients engaged in the program. A challenge to the effectiveness of programs like Noom is user engagement - users often quit the program early, or do not use it as intended [DETAILED PROBLEM]. What novel solutions can you come up with to improve user engagement with Noom? [ASK]*

PAMS Technical Requirements: *You are a software architect working in a medical school and designing a randomized control trial for pre-diabetic patients [ROLE]. For the trial we need you to design a system that collects weekly patient data from a Noom Web service endpoint and determines patient engagement based on patient interaction with Noom. Every week each patient will be assigned a fail/pass score based on if they meet the CDC gold standard guidelines for type 2 diabetes prevention. These scores will be aggregated every three weeks to determine adherence trends. Based on these trends, patients will receive a behavioral science SMS on their provider's behalf to promote engagement with the overall goal is to increase adherence to the Noom program [DETAILED PROBLEM]. Please produce a high-level design for this system so we can start a conversation with a team of software developers [ASK].*

Results

PAMS Concept and User Experience Generation

Overview

Core components of the PAMS intervention were conceptualized and designed via an underlying behavior change theory, design principles and personas, and a message content library.

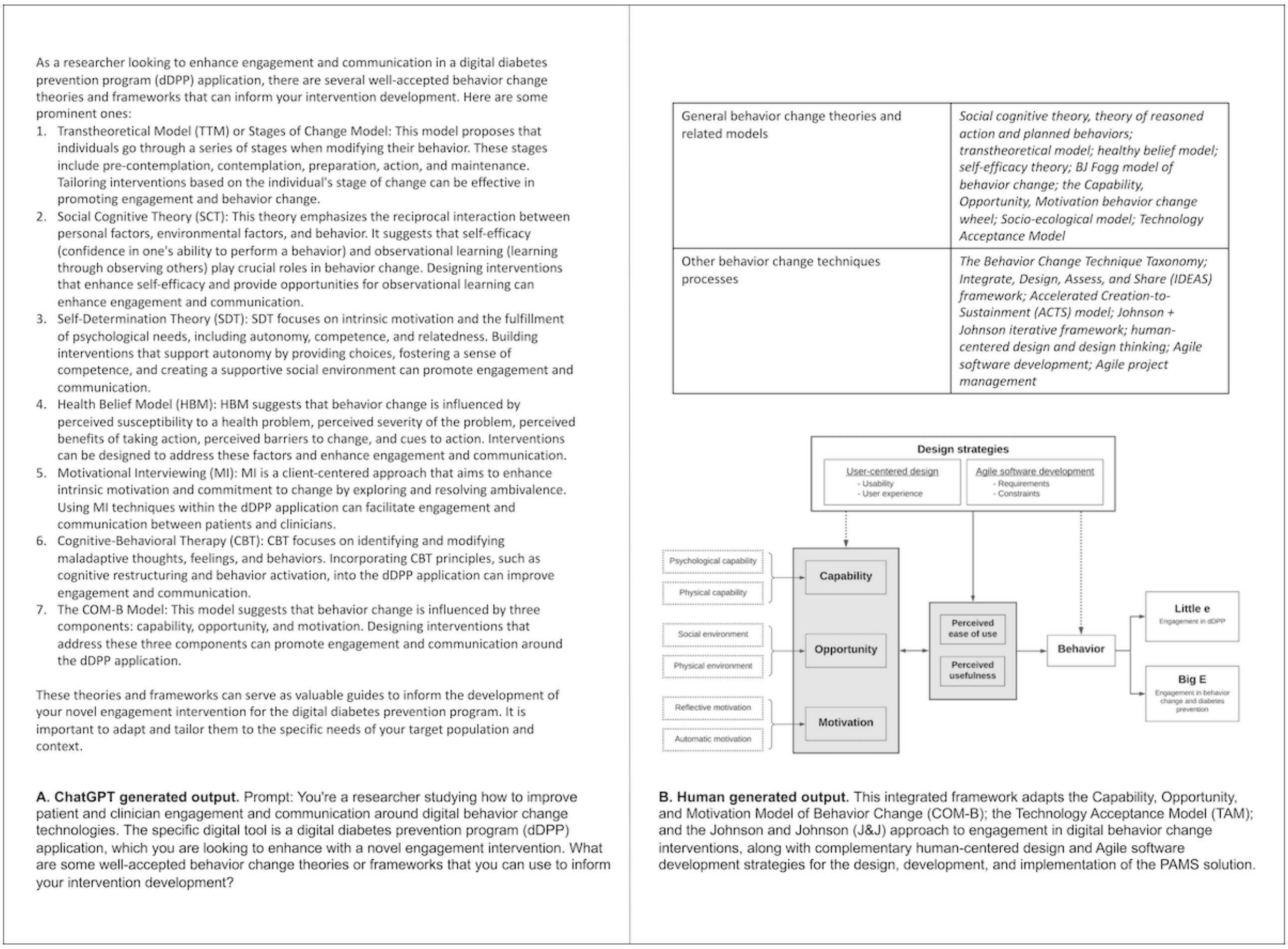
Underpinning Behavior Change Theory and Approach

Human-Generated Solution

Leveraging behavior change literature review and interviews with behavior change theory content experts (n=4), the research team initially identified ten unique behavior change theories and six process models that were considered to be an appropriate fit for the aims of the overall intervention. A unique model was developed that captured (1) the relevant underlying behavior change theory, (2) implementation strategies, and (3) unique contexts of the technology environment (Figure 2A).

XSL•FO
RenderX

Figure 2. Underpinning behavior change theory and approach outcome of ChatGPT vs human-generated output. ACTS: Accelerated Creation to Sustainment; BJ: Brian Jeffrey; CBT: cognitive behavioral therapy; COM-B: capability, opportunity, and motivation model of behavior change; dDPP: digital diabetes prevention program; HBM: health belief model; IDEAS: Integrate, Design, Assess, and Share; J&J: Johnson and Johnson; MI: motivational interviewing; SCT: social cognitive theory; SDT: self-determination theory; TTM: transtheoretical model.



As a researcher looking to enhance engagement and communication in a digital diabetes prevention program (dDPP) application, there are several well-accepted behavior change theories and frameworks that can inform your intervention development. Here are some prominent ones:

1. Transtheoretical Model (TTM) or Stages of Change Model: This model proposes that individuals go through a series of stages when modifying their behavior. These stages include pre-contemplation, contemplation, preparation, action, and maintenance. Tailoring interventions based on the individual's stage of change can be effective in promoting engagement and behavior change.
2. Social Cognitive Theory (SCT): This theory emphasizes the reciprocal interaction between personal factors, environmental factors, and behavior. It suggests that self-efficacy (confidence in one's ability to perform a behavior) and observational learning (learning through observing others) play crucial roles in behavior change. Designing interventions that enhance self-efficacy and provide opportunities for observational learning can enhance engagement and communication.
3. Self-Determination Theory (SDT): SDT focuses on intrinsic motivation and the fulfillment of psychological needs, including autonomy, competence, and relatedness. Building interventions that support autonomy by providing choices, fostering a sense of competence, and creating a supportive social environment can promote engagement and communication.
4. Health Belief Model (HBM): HBM suggests that behavior change is influenced by perceived susceptibility to a health problem, perceived severity of the problem, perceived benefits of taking action, perceived barriers to change, and cues to action. Interventions can be designed to address these factors and enhance engagement and communication.
5. Motivational Interviewing (MI): MI is a client-centered approach that aims to enhance intrinsic motivation and commitment to change by exploring and resolving ambivalence. Using MI techniques within the dDPP application can facilitate engagement and communication between patients and clinicians.
6. Cognitive-Behavioral Therapy (CBT): CBT focuses on identifying and modifying maladaptive thoughts, feelings, and behaviors. Incorporating CBT principles, such as cognitive restructuring and behavior activation, into the dDPP application can improve engagement and communication.
7. The COM-B Model: This model suggests that behavior change is influenced by three components: capability, opportunity, and motivation. Designing interventions that address these three components can promote engagement and communication around the dDPP application.

These theories and frameworks can serve as valuable guides to inform the development of your novel engagement intervention for the digital diabetes prevention program. It is important to adapt and tailor them to the specific needs of your target population and context.

A. ChatGPT generated output. Prompt: You're a researcher studying how to improve patient and clinician engagement and communication around digital behavior change technologies. The specific digital tool is a digital diabetes prevention program (dDPP) application, which you are looking to enhance with a novel engagement intervention. What are some well-accepted behavior change theories or frameworks that you can use to inform your intervention development?

General behavior change theories and related models	*Social cognitive theory, theory of reasoned action and planned behaviors; transtheoretical model; healthy belief model; self-efficacy theory; BJ Fogg model of behavior change; the Capability, Opportunity, Motivation behavior change wheel; Socio-ecological model; Technology Acceptance Model*
Other behavior change techniques processes	*The Behavior Change Technique Taxonomy; Integrate, Design, Assess, and Share (IDEAS) framework; Accelerated Creation-to-Sustainment (ACTS) model; Johnson + Johnson iterative framework; human-centered design and design thinking; Agile software development; Agile project management*

B. Human generated output. This integrated framework adapts the Capability, Opportunity, and Motivation Model of Behavior Change (COM-B); the Technology Acceptance Model (TAM); and the Johnson and Johnson (J&J) approach to engagement in digital behavior change interventions, along with complementary human-centered design and Agile software development strategies for the design, development, and implementation of the PAMS solution.

GenAI Solution

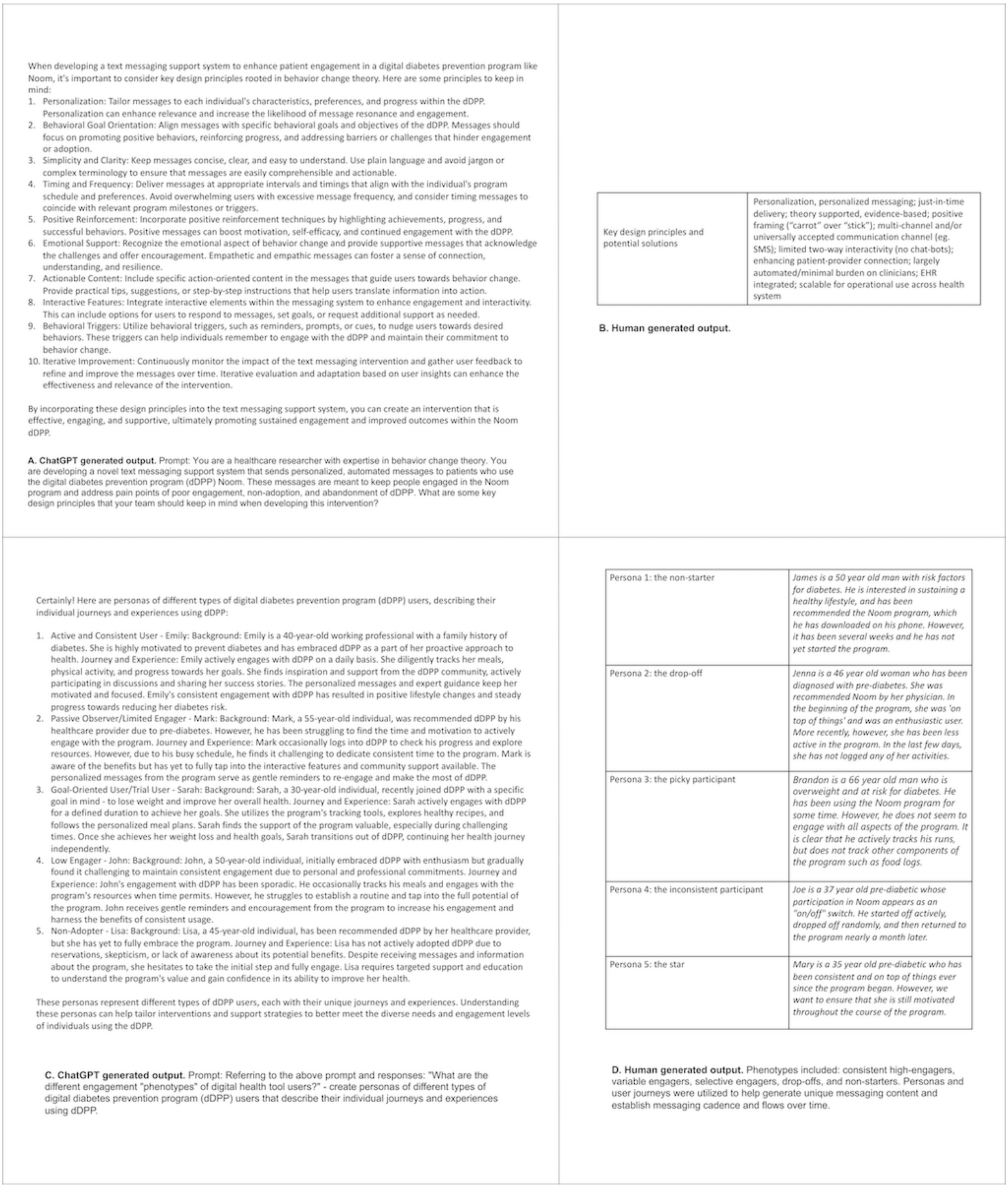
When prompted, ChatGPT identified seven relevant well-accepted behavior change theories and frameworks to inform a “dDPP support intervention” (Figure 2B). It did not provide information on the transtheoretical domains framework, or the taxonomy of behavior change techniques, but when prompted on these, identified both as acceptable strategies for use.

User Experience: Design Principles, Personas, and Messaging Content

Human-Generated Solution

The research team used a human-centered design approach to identify key design principles, defined as the set of considerations that form the basis of the PAMS product (Figure 3B). These were developed from insights gathered via a review of relevant digital behavior change research, consultation with content and theoretical experts in digital health and implementation science (n=3), and two group interviews (n=9). From these insights, five relevant fictional personas were designed to capture the various phenotypes of user engagement with the commercial dDPP, along with unique user journeys developed to describe their projected engagement with the program over time (Figure 3D). Overall, over 193 unique messages were developed, each grounded by a relevant behavior change technique and tailored to an individual phenotype’s user journey. These elements were continuously revisited and refined during the testing phases of the dDPP research. This included a 6-month near-live user testing phase consisting of nine patients engaging with various iterations of the PAMS prototype, and a 12-month live single-arm pilot phase consisting of 25 patients using PAMS-beta with the commercial dDPP platform.

Figure 3. User experience: design principles, personas, and messaging content outcome of ChatGPT vs human-generated output. dDPP: digital diabetes prevention program; EHR: electronic health record.



When developing a text messaging support system to enhance patient engagement in a digital diabetes prevention program like Noom, it's important to consider key design principles rooted in behavior change theory. Here are some principles to keep in mind:

1. Personalization: Tailor messages to each individual's characteristics, preferences, and progress within the dDPP. Personalization can enhance relevance and increase the likelihood of message resonance and engagement.
2. Behavioral Goal Orientation: Align messages with specific behavioral goals and objectives of the dDPP. Messages should focus on promoting positive behaviors, reinforcing progress, and addressing barriers or challenges that hinder engagement or adoption.
3. Simplicity and Clarity: Keep messages concise, clear, and easy to understand. Use plain language and avoid jargon or complex terminology to ensure that messages are easily comprehensible and actionable.
4. Timing and Frequency: Deliver messages at appropriate intervals and timings that align with the individual's program schedule and preferences. Avoid overwhelming users with excessive message frequency, and consider timing messages to coincide with relevant program milestones or triggers.
5. Positive Reinforcement: Incorporate positive reinforcement techniques by highlighting achievements, progress, and successful behaviors. Positive messages can boost motivation, self-efficacy, and continued engagement with the dDPP.
6. Emotional Support: Recognize the emotional aspect of behavior change and provide supportive messages that acknowledge the challenges and offer encouragement. Empathetic and empathic messages can foster a sense of connection, understanding, and resilience.
7. Actionable Content: Include specific action-oriented content in the messages that guide users towards behavior change. Provide practical tips, suggestions, or step-by-step instructions that help users translate information into action.
8. Interactive Features: Integrate interactive elements within the messaging system to enhance engagement and interactivity. This can include options for users to respond to messages, set goals, or request additional support as needed.
9. Behavioral Triggers: Utilize behavioral triggers, such as reminders, prompts, or cues, to nudge users towards desired behaviors. These triggers can help individuals remember to engage with the dDPP and maintain their commitment to behavior change.
10. Iterative Improvement: Continuously monitor the impact of the text messaging intervention and gather user feedback to refine and improve the messages over time. Iterative evaluation and adaptation based on user insights can enhance the effectiveness and relevance of the intervention.

By incorporating these design principles into the text messaging support system, you can create an intervention that is effective, engaging, and supportive, ultimately promoting sustained engagement and improved outcomes within the Noom dDPP.

A. ChatGPT generated output. Prompt: You are a healthcare researcher with expertise in behavior change theory. You are developing a novel text messaging support system that sends personalized, automated messages to patients who use the digital diabetes prevention program (dDPP) Noom. These messages are meant to keep people engaged in the Noom program and address pain points of poor engagement, non-adoption, and abandonment of dDPP. What are some key design principles that your team should keep in mind when developing this intervention?

Key design principles and potential solutions	Personalization, personalized messaging; just-in-time delivery; theory supported, evidence-based; positive framing (“carrot” over “stick”); multi-channel and/or universally accepted communication channel (eg. SMS); limited two-way interactivity (no chat-bots); enhancing patient-provider connection; largely automated/minimal burden on clinicians; EHR integrated; scalable for operational use across health system

B. Human generated output.

Certainly! Here are personas of different types of digital diabetes prevention program (dDPP) users, describing their individual journeys and experiences using dDPP:

1. Active and Consistent User - Emily: Background: Emily is a 40-year-old working professional with a family history of diabetes. She is highly motivated to prevent diabetes and has embraced dDPP as a part of her proactive approach to health. Journey and Experience: Emily actively engages with dDPP on a daily basis. She diligently tracks her meals, physical activity, and progress towards her goals. She finds inspiration and support from the dDPP community, actively participating in discussions and sharing her success stories. The personalized messages and expert guidance keep her motivated and focused. Emily's consistent engagement with dDPP has resulted in positive lifestyle changes and steady progress towards reducing her diabetes risk.
2. Passive Observer/Limited Engager - Mark: Background: Mark, a 55-year-old individual, was recommended dDPP by his healthcare provider due to pre-diabetes. However, he has been struggling to find the time and motivation to actively engage with the program. Journey and Experience: Mark occasionally logs into dDPP to check his progress and explore resources. However, due to his busy schedule, he finds it challenging to dedicate consistent time to the program. Mark is aware of the benefits but has yet to fully tap into the interactive features and community support available. The personalized messages from the program serve as gentle reminders to re-engage and make the most of dDPP.
3. Goal-Oriented User/Trial User - Sarah: Background: Sarah, a 30-year-old individual, recently joined dDPP with a specific goal in mind - to lose weight and improve her overall health. Journey and Experience: Sarah actively engages with dDPP for a defined duration to achieve her goals. She utilizes the program's tracking tools, explores healthy recipes, and follows the personalized meal plans. Sarah finds the support of the program valuable, especially during challenging times. Once she achieves her weight loss and health goals, Sarah transitions out of dDPP, continuing her health journey independently.
4. Low Engager - John: Background: John, a 50-year-old individual, initially embraced dDPP with enthusiasm but gradually found it challenging to maintain consistent engagement due to personal and professional commitments. Journey and Experience: John's engagement with dDPP has been sporadic. He occasionally tracks his meals and engages with the program's resources when time permits. However, he struggles to establish a routine and tap into the full potential of the program. John receives gentle reminders and encouragement from the program to increase his engagement and harness the benefits of consistent usage.
5. Non-Adopter - Lisa: Background: Lisa, a 45-year-old individual, has been recommended dDPP by her healthcare provider, but she has yet to fully embrace the program. Journey and Experience: Lisa has not actively adopted dDPP due to reservations, skepticism, or lack of awareness about its potential benefits. Despite receiving messages and information about the program, she hesitates to take the initial step and fully engage. Lisa requires targeted support and education to understand the program's value and gain confidence in its ability to improve her health.

These personas represent different types of dDPP users, each with their unique journeys and experiences. Understanding these personas can help tailor interventions and support strategies to better meet the diverse needs and engagement levels of individuals using the dDPP.

C. ChatGPT generated output. Prompt: Referring to the above prompt and responses: "What are the different engagement "phenotypes" of digital health tool users?" - create personas of different types of digital diabetes prevention program (dDPP) users that describe their individual journeys and experiences using dDPP.

Persona 1: the non-starter	*James is a 50 year old man with risk factors for diabetes. He is interested in sustaining a healthy lifestyle, and has been recommended the Noom program, which he has downloaded on his phone. However, it has been several weeks and he has not yet started the program.*
Persona 2: the drop-off	*Jenna is a 46 year old woman who has been diagnosed with pre-diabetes. She was recommended Noom by her physician. In the beginning of the program, she was 'on top of things' and was an enthusiastic user. More recently, however, she has been less active in the program. In the last few days, she has not logged any of her activities.*
Persona 3: the picky participant	*Brandon is a 66 year old man who is overweight and at risk for diabetes. He has been using the Noom program for some time. However, he does not seem to engage with all aspects of the program. It is clear that he actively tracks his runs, but does not track other components of the program such as food logs.*
Persona 4: the inconsistent participant	*Joe is a 37 year old pre-diabetic whose participation in Noom appears as an "on/off" switch. He started off actively, dropped off randomly, and then returned to the program nearly a month later.*
Persona 5: the star	*Mary is a 35 year old pre-diabetic who has been consistent and on top of things ever since the program began. However, we want to ensure that she is still motivated throughout the course of the program.*

D. Human generated output. Phenotypes included: consistent high-engagers, variable engagers, selective engagers, drop-offs, and non-starters. Personas and user journeys were utilized to help generate unique messaging content and establish messaging cadence and flows over time.

GenAI Solution

ChatGPT was prompted from multiple perspectives (researcher, clinician, and patient) to identify key design principles (Figure 3A) and sample solutions for the PAMS intervention. It also provided common engagement phenotypes for digital health tool users, based on patterns of use, frequency, duration, and “other elements.” Of note, nonadopters were not identified within the initial round of phenotypes. ChatGPT also developed personas for each of the identified engagement phenotypes, including persona names, backgrounds, and individual journeys. ChatGPT was able to produce five to ten unique messages targeted toward each phenotype and to adapt these messages based on various additional prompts. The user types or personas generated by ChatGPT are consistent with the human-generated users and cover all the phenotypes identified in our previous research (eg, mapping to a specific behavior change technique and reflecting a key design principle; Figure 3C).

XSL•FO
RenderX

PAMS Technical Development

Overview

The technical development includes a PAMS requirements document and architectural design and code.

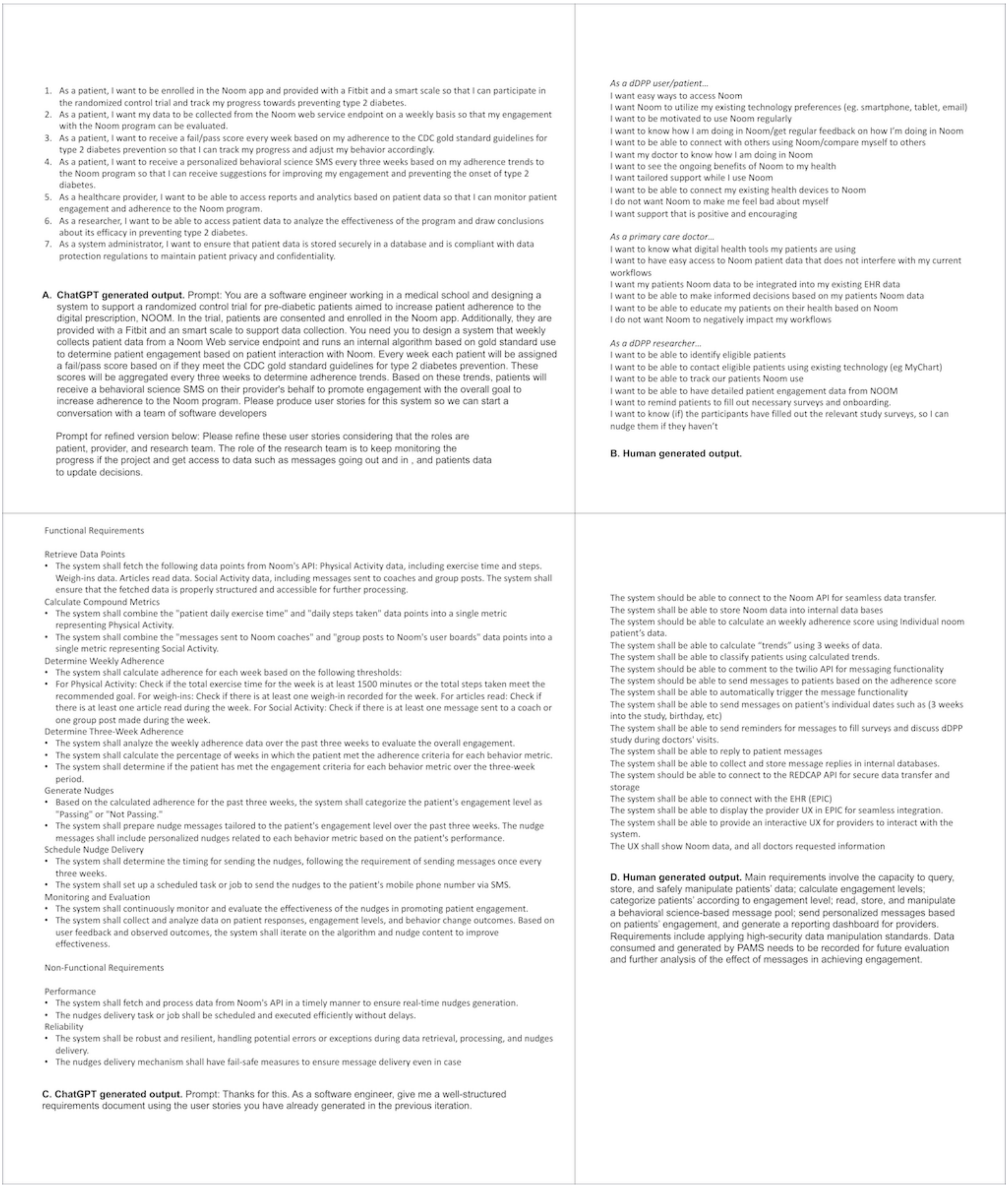
Technical Requirements (User Stories)

Human-Generated Solution

Following the data collection and intervention design period, we created, as a team, a series of user stories (Figure 4B) which were followed by system requirements to describe the intended use cases, features, and challenges of the proposed PAMS software. Initial system requirements represent the “minimum viable product” that was developed, piloted, and further refined (Figure 4D). Our development team followed software engineering principles to generate the requirements document.

XSL•FO
RenderX

Figure 4. User stories and technical requirements outcome of ChatGPT vs human-generated outputs. API: application programming interface; CDC: Centers for Disease Control and Prevention; EHR: electronic health record; PAMS: personalized automatic messaging system; REDCap: Research Electronic Data Capture; UX: user interface.



1. As a patient, I want to be enrolled in the Noom app and provided with a Fitbit and a smart scale so that I can participate in the randomized control trial and track my progress towards preventing type 2 diabetes.
2. As a patient, I want my data to be collected from the Noom web service endpoint on a weekly basis so that my engagement with the Noom program can be evaluated.
3. As a patient, I want to receive a fail/pass score every week based on my adherence to the CDC gold standard guidelines for type 2 diabetes prevention so that I can track my progress and adjust my behavior accordingly.
4. As a patient, I want to receive a personalized behavioral science SMS every three weeks based on my adherence trends to the Noom program so that I can receive suggestions for improving my engagement and preventing the onset of type 2 diabetes.
5. As a healthcare provider, I want to be able to access reports and analytics based on patient data so that I can monitor patient engagement and adherence to the Noom program.
6. As a researcher, I want to be able to access patient data to analyze the effectiveness of the program and draw conclusions about its efficacy in preventing type 2 diabetes.
7. As a system administrator, I want to ensure that patient data is stored securely in a database and is compliant with data protection regulations to maintain patient privacy and confidentiality.

A. **ChatGPT generated output.** Prompt: You are a software engineer working in a medical school and designing a system to support a randomized control trial for pre-diabetic patients aimed to increase patient adherence to the digital prescription, NOOM. In the trial, patients are consented and enrolled in the Noom app. Additionally, they are provided with a Fitbit and an smart scale to support data collection. You need you to design a system that weekly collects patient data from a Noom Web service endpoint and runs an internal algorithm based on gold standard use to determine patient engagement based on patient interaction with Noom. Every week each patient will be assigned a fail/pass score based on if they meet the CDC gold standard guidelines for type 2 diabetes prevention. These scores will be aggregated every three weeks to determine adherence trends. Based on these trends, patients will receive a behavioral science SMS on their provider's behalf to promote engagement with the overall goal to increase adherence to the Noom program. Please produce user stories for this system so we can start a conversation with a team of software developers

Prompt for refined version below: Please refine these user stories considering that the roles are patient, provider, and research team. The role of the research team is to keep monitoring the progress if the project and get access to data such as messages going out and in , and patients data to update decisions.

As a dDPP user/patient…
I want easy ways to access Noom
I want Noom to utilize my existing technology preferences (eg. smartphone, tablet, email)
I want to be motivated to use Noom regularly
I want to know how I am doing in Noom/get regular feedback on how I'm doing in Noom
I want to be able to connect with others using Noom/compare myself to others
I want my doctor to know how I am doing in Noom
I want to see the ongoing benefits of Noom to my health
I want tailored support while I use Noom
I want to be able to connect my existing health devices to Noom
I do not want Noom to make me feel bad about myself
I want support that is positive and encouraging

As a primary care doctor…
I want to know what digital health tools my patients are using
I want to have easy access to Noom patient data that does not interfere with my current workflows
I want my patients Noom data to be integrated into my existing EHR data
I want to be able to make informed decisions based on my patients Noom data
I want to be able to educate my patients on their health based on Noom
I do not want Noom to negatively impact my workflows

As a dDPP researcher…
I want to be able to identify eligible patients
I want to be able to contact eligible patients using existing technology (eg MyChart)
I want to be able to track our patients Noom use
I want to be able to have detailed patient engagement data from NOOM
I want to remind patients to fill out necessary surveys and onboarding.
I want to know (if) the participants have filled out the relevant study surveys, so I can nudge them if they haven't

B. Human generated output.

Functional Requirements

Retrieve Data Points
- The system shall fetch the following data points from Noom's API: Physical Activity data, including exercise time and steps. Weigh-ins data. Articles read data. Social Activity data, including messages sent to coaches and group posts. The system shall ensure that the fetched data is properly structured and accessible for further processing.

Calculate Compound Metrics
- The system shall combine the "patient daily exercise time" and "daily steps taken" data points into a single metric representing Physical Activity.
- The system shall combine the "messages sent to Noom coaches" and "group posts to Noom's user boards" data points into a single metric representing Social Activity.

Determine Weekly Adherence
- The system shall calculate adherence for each week based on the following thresholds:
- For Physical Activity: Check if the total exercise time for the week is at least 1500 minutes or the total steps taken meet the recommended goal. For weigh-ins: Check if there is at least one weigh-in recorded for the week. For articles read: Check if there is at least one article read during the week. For Social Activity: Check if there is at least one message sent to a coach or one group post made during the week.

Determine Three-Week Adherence
- The system shall analyze the weekly adherence data over the past three weeks to evaluate the overall engagement.
- The system shall calculate the percentage of weeks in which the patient met the adherence criteria for each behavior metric.
- The system shall determine if the patient has met the engagement criteria for each behavior metric over the three-week period.

Generate Nudges
- Based on the calculated adherence for the past three weeks, the system shall categorize the patient's engagement level as "Passing" or "Not Passing."
- The system shall prepare nudge messages tailored to the patient's engagement level over the past three weeks. The nudge messages shall include personalized nudges related to each behavior metric based on the patient's performance.

Schedule Nudge Delivery
- The system shall determine the timing for sending the nudges, following the requirement of sending messages once every three weeks.
- The system shall set up a scheduled task or job to send the nudges to the patient's mobile phone number via SMS.

Monitoring and Evaluation
- The system shall continuously monitor and evaluate the effectiveness of the nudges in promoting patient engagement.
- The system shall collect and analyze data on patient responses, engagement levels, and behavior change outcomes. Based on user feedback and observed outcomes, the system shall iterate on the algorithm and nudge content to improve effectiveness.

Non-Functional Requirements

Performance
- The system shall fetch and process data from Noom's API in a timely manner to ensure real-time nudges generation.
- The nudges delivery task or job shall be scheduled and executed efficiently without delays.

Reliability
- The system shall be robust and resilient, handling potential errors or exceptions during data retrieval, processing, and nudges delivery.
- The nudges delivery mechanism shall have fail-safe measures to ensure message delivery even in case

C. ChatGPT generated output. Prompt: Thanks for this. As a software engineer, give me a well-structured requirements document using the user stories you have already generated in the previous iteration.

The system should be able to connect to the Noom API for seamless data transfer.
The system shall be able to store Noom data into internal data bases
The system should be able to calculate an weekly adherence score using Individual noom patient's data.
The system shall be able to calculate "trends" using 3 weeks of data.
The system shall be able to classify patients using calculated trends.
The system should be able to comment to the twilio API for messaging functionality
The system should be able to send messages to patients based on the adherence score
The system shall be able to automatically trigger the message functionality
The system shall be able to send messages on patient's individual dates such as (3 weeks into the study, birthday, etc)
The system shall be able to send reminders for messages to fill surveys and discuss dDPP study during doctors' visits.
The system shall be able to reply to patient messages
The system shall be able to collect and store message replies in internal databases.
The system should be able to connect to the REDCAP API for secure data transfer and storage
The system shall be able to connect with the EHR (EPIC)
The system shall be able to display the provider UX in EPIC for seamless integration.
The system shall be able to provide an interactive UX for providers to interact with the system.
The UX shall show Noom data, and all doctors requested information

D. Human generated output. Main requirements involve the capacity to query, store, and safely manipulate patients' data; calculate engagement levels; categorize patients' according to engagement level; read, store, and manipulate a behavioral science-based message pool; send personalized messages based on patients' engagement, and generate a reporting dashboard for providers. Requirements include applying high-security data manipulation standards. Data consumed and generated by PAMS needs to be recorded for future evaluation and further analysis of the effect of messages in achieving engagement.

GenAI Solution

We used the output of the "feature construction phase" to inform the GenAI output for requirements. During the initial stages of the prompting phase, we refrained from suggesting solutions, allowing ChatGPT to generate potential solutions autonomously. We reviewed and evaluated these outputs, eliminating impractical or incompatible solution paths that did not align with the intentions or capabilities of our team. Once we reached a satisfactory outcome but faced uncertainty regarding the next steps, we instructed ChatGPT to assume a different "personality" (eg, software architect) and used the previous outputs as a foundation for the new role's initial prompts. Throughout this process, we encouraged each "personality" to seek clarifications by asking questions and provided feedback without biasing toward any predetermined solution. We repeated this process at least four times for each personality type, engaging in a

XSL•FO
RenderX

back-and-forth roleplay with multiple personalities (researcher, architect, and developer), transitioning to a different personality when it became evident that the current one could no longer progress without additional feedback (Figures 4A and 4C).

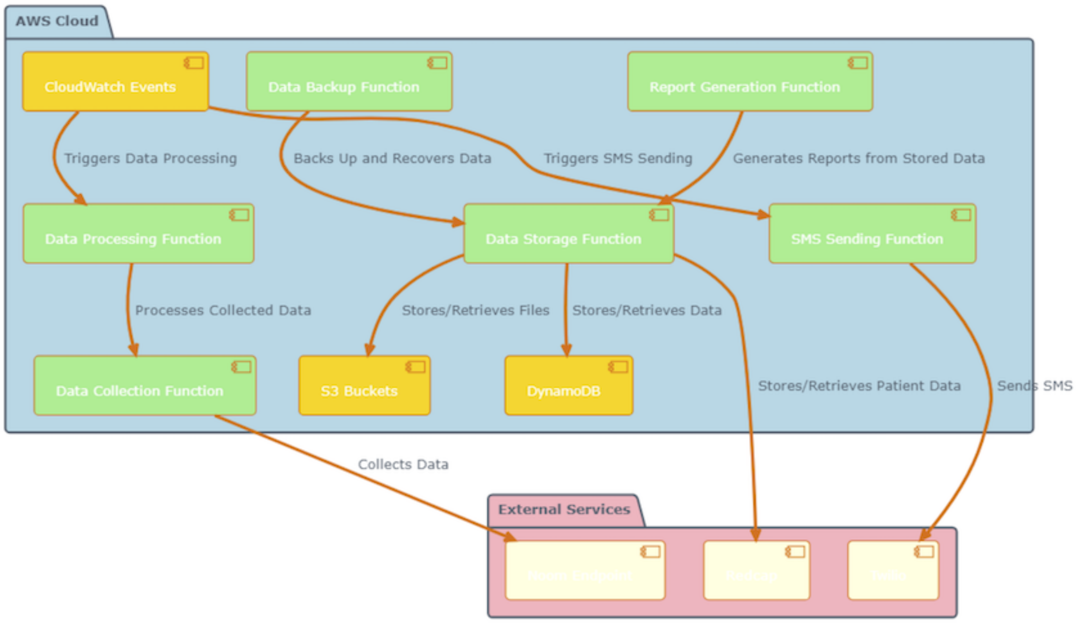
Architectural Design

Human-Generated Solution

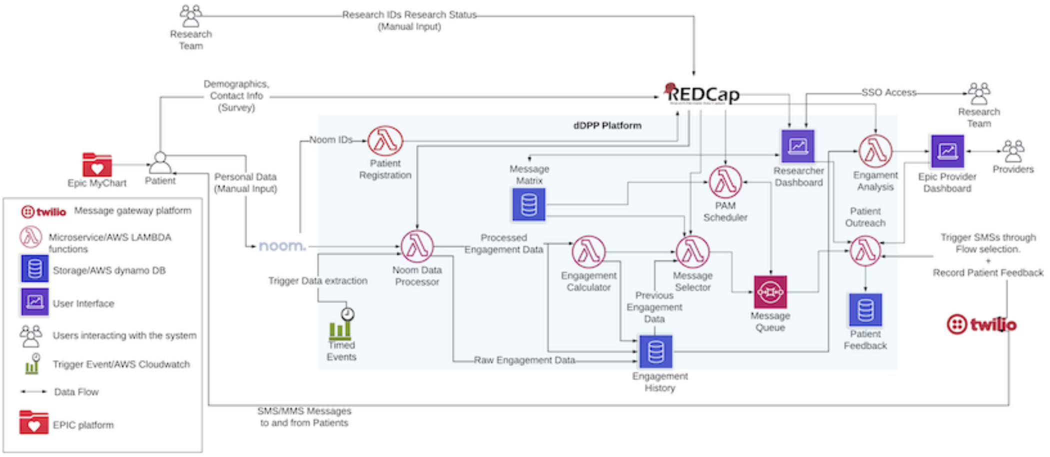
After the requirement phase, our software development team developed the PAMS architectural diagram, which is a graphical representation of the system that includes (1) a set of components (eg, a database and computational modules) that will perform a function required by the system; (2) the set of connectors that will help in coordination, communication, and cooperation between the components; and (3) conditions for how components can be integrated to form the system (Figure 5B).

XSL•FO
RenderX

Figure 5. Architectural design for PAMS. ChatGPT vs human-generated output. AWS: Amazon Web Services; dDPP: digital diabetes prevention program; PAMS: personalized automatic messaging system; REDCap: Research Electronic Data Capture.



A. ChatGPT generated output. Prompt: Thanks for that. With the generated requirements, I will like for you to provide the necessary architectural diagrams for the system. Please consider the following: to develop the system, we want to use AWS and a micro-services approach, also patient data collected from the patients will be stored in RedCap, and we will use Twilio as the SMS gateway to deliver the text messages. Please redo the diagrams using this information.



B. Human generated output. Internal PAMS functionality includes data collection, data analysis, message selection, and message delivery processes, via a modular design where each module is independent and governs a specific activity or functionality within the system. The blue-highlighted area in the figure represents PAMS itself, to include the associated modules, we use symbols. Components outside the blue-highlighted area are external systems and operations that interact with PAMS. These include external applications and research activities that produce data to support PAMS' functionalities or utilize data coming from PAMS. Arrows in the figure show the directions of data flows.

GenAI Solution

For the GenAI-generated architectural design, we leveraged the outputs of the requirement phase and the available ChatGPT plugins to designate the GenAI model as a software engineer and proceeded to develop an architectural diagram. During this process, we engaged in iterative prompting and provided explicit instructions to ChatGPT, specifying the use of Amazon Web Services (AWS) for development, integration of external systems such as Twilio (Twilio Inc) and REDCap (Research Electronic Data Capture; Vanderbilt University), and the adoption of a microservice approach to facilitate the efforts of our development team (Figure 5A).

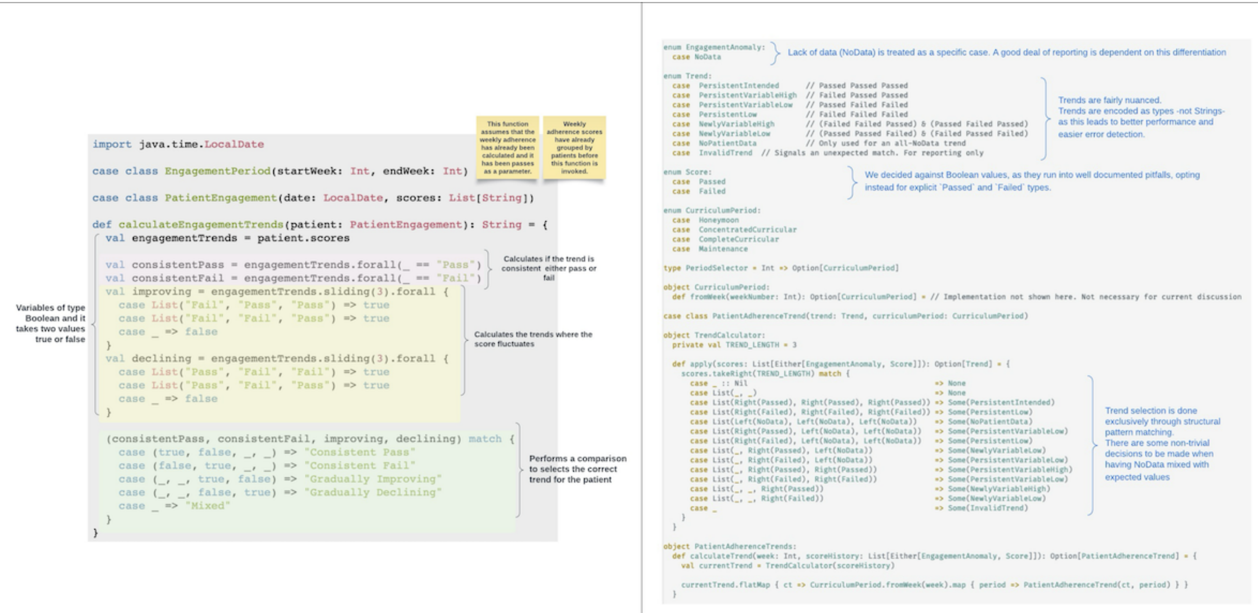
XSL•FO
RenderX

Code

Human-Generated Solution

PAMS components include several lambda functions that execute its engagement or adherence algorithm, messaging, and data manipulation functionalities. Most of the functions are coded and developed using Python (Python Software Foundation) and Scala (École Polytechnique Fédérale Lausanne) as programming languages. AWS was used for the development of PAMS [12]. Our developers followed our microservice approach design using an event-driven model [13,14]. The main components of PAMS are AWS lambda functions which are triggered by different events such as updates to S3 buckets, modifications on DynamoDB (AWS) tables, or CloudWatch (AWS) events. External interactions of PAMS use application programming interface calls, which secure effective data transfer (Figure 6B).

Figure 6. Code for the function that calculates patient adherence and engagement trends. ChatGPT vs human-generated outputs.



```
import java.time.LocalDate

case class EngagementPeriod(startWeek: Int, endWeek: Int)

case class PatientEngagement(date: LocalDate, scores: List[String])

def calculateEngagementTrends(patient: PatientEngagement): String = {
  val engagementTrends = patient.scores

  val consistentPass = engagementTrends.forall(_ == "Pass")
  val consistentFail = engagementTrends.forall(_ == "Fail")
  val improving = engagementTrends.sliding(3).forall {
    case List("Fail", "Pass", "Pass") => true
    case List("Fail", "Fail", "Pass") => true
    case _ => false
  }
  val declining = engagementTrends.sliding(3).forall {
    case List("Pass", "Fail", "Fail") => true
    case List("Pass", "Fail", "Pass") => true
    case _ => false
  }

  (consistentPass, consistentFail, improving, declining) match {
    case (true, false, _, _) => "Consistent Pass"
    case (false, true, _, _) => "Consistent Fail"
    case (_, _, true, false) => "Gradually Improving"
    case (_, _, false, true) => "Gradually Declining"
    case _ => "Mixed"
  }
}
```

This function assumes that the weekly adherence has already been calculated and it has been passes as a parameter.

Weekly adherence scores have already grouped by patients before this function is invoked.

Calculates if the trend is consistent either pass or fail

Variables of type Boolean and it takes two values true or false

Calculates the trends where the score fluctuates

Performs a comparison to selects the correct trend for the patient

A. ChatGPT generated output. Prompt: You are an experienced software developer tasked with writing code for the business logic. Write code in Scala for the function responsible for calculating engagement trends for patients every three weeks. Assume all other functionality is being worked on independently and mock calls to services and data sources as needed.

```
enum EngagementAnomaly:
  case NoData

enum Trend:
  case  PersistentIntended      // Passed Passed Passed
  case  PersistentVariableHigh  // Failed Passed Passed
  case  PersistentVariableLow   // Passed Failed Failed
  case  PersistentLow           // Failed Failed Failed
  case  NewlyVariableHigh       // (Failed Failed Passed) & (Passed Failed Passed)
  case  NewlyVariableLow        // (Passed Passed Failed) & (Failed Passed Failed)
  case  NoPatientData           // Only used for an all-NoData trend
  case  InvalidTrend  // Signals an unexpected match. For reporting only

enum Score:
  case  Passed
  case  Failed

enum CurriculumPeriod:
  case  Honeymoon
  case  ConcentratedCurricular
  case  CompleteCurricular
  case  Maintenance

type PeriodSelector = Int => Option[CurriculumPeriod]

object CurriculumPeriod:
  def fromWeek(weekNumber: Int): Option[CurriculumPeriod] = // Implementation not shown here. Not necessary for current discussion

case class PatientAdherenceTrend(trend: Trend, curriculumPeriod: CurriculumPeriod)

object TrendCalculator:
  private val TREND_LENGTH = 3

  def apply(scores: List[Either[EngagementAnomaly, Score]]): Option[Trend] = {
    scores.takeRight(TREND_LENGTH) match {
      case _ :: Nil                                           => None
      case List(_, _)                                         => None
      case List(Right(Passed), Right(Passed), Right(Passed)) => Some(PersistentIntended)
      case List(Right(Failed), Right(Failed), Right(Failed)) => Some(PersistentLow)
      case List(Left(NoData), Left(NoData), Left(NoData))    => Some(NoPatientData)
      case List(Right(Passed), Left(NoData), Left(NoData))   => Some(PersistentVariableLow)
      case List(Right(Failed), Left(NoData), Left(NoData))   => Some(PersistentLow)
      case List(_, Right(Passed), Left(NoData))               => Some(NewlyVariableLow)
      case List(_, Right(Failed), Left(NoData))               => Some(PersistentVariableLow)
      case List(_, Right(Passed), Right(Passed))              => Some(PersistentVariableHigh)
      case List(_, Right(Failed), Right(Failed))              => Some(PersistentVariableLow)
      case List(_, _, Right(Passed))                          => Some(NewlyVariableHigh)
      case List(_, _, Right(Failed))                          => Some(NewlyVariableLow)
      case _                                                  => Some(InvalidTrend)
    }
  }

object PatientAdherenceTrends:
  def calculateTrend(week: Int, scoreHistory: List[Either[EngagementAnomaly, Score]]): Option[PatientAdherenceTrend] = {
    val currentTrend = TrendCalculator(scoreHistory)

    currentTrend.flatMap { ct => CurriculumPeriod.fromWeek(week).map { period => PatientAdherenceTrend(ct, period) } }
  }
```

Lack of data (NoData) is treated as a specific case. A good deal of reporting is dependent on this differentiation

Trends are fairly nuanced.
Trends are encoded as types -not Strings- as this leads to better performance and easier error detection.

We decided against Boolean values, as they run into well documented pitfalls, opting instead for explicit `Passed` and `Failed` types.

Trend selection is done exclusively through structural pattern matching.
There are some non-trivial decisions to be made when having NoData mixed with expected values

B. Human generated output. The code was generated by our developer following traditional software engineering principles and standardized code development styles.

GenAI Solution

To facilitate the generation of the coded solution using ChatGPT, we assigned the role of a software engineer to the model and specifically requested it to generate Scala code for a specific functionality, namely the "calculate engagement trends" function. Consistent with the iterative nature of the GenAI-based software development process, we engaged in a back-and-forth interaction with ChatGPT, iterating over the prompt and its output while providing expert guidance to ensure optimal results. While allowing ChatGPT to generate free text, we evaluated each output for accuracy and adherence to the desired specifications (Figure 6A).

Internal Review of Human Vs GenAI Outputs

The 11 evaluators participated in the output review process. All had familiarity or prior exposure to the original PAMS intervention. Overall, evaluators rated the ChatGPT-produced outputs as positive for the theoretical background and design phase in terms of understandability, usability, novelty, relevance, and efficiency. For these two components, the question about completeness showed the most variability with divided opinion among "agree" and "disagree" and the bias was mostly categorized as "neither agree nor disagree." For the first part of the technical development (user stories and requirement documents), most of the raters found the ChatGPT output positive in terms of understandability, usability, and relevance. In terms of completeness and novelty, requirements were better rated than the user stories which represent an interesting output since requirements are derived from the user stories. We hypothesize that our raters were expecting better user stories, but once these were defined, they considered ChatGPT to be effective at turning these into the requirements. In terms of bias, similar to the theoretical background and design phase, the most popular answer was "neither agree nor disagree." For the more technical pieces of the development that required software engineering knowledge, specifically the architectural diagram and code elements, results showed the highest N/A responses. These higher levels of N/As were associated with lower levels of expertise (eg, coding experience) since only 2 of the 11 evaluators had computer science backgrounds. However, the overall score excluding the N/As was positive for the technical component.

Discussion

Results Summary

This study leveraged ChatGPT-4 to recreate content features and software development of PAMS. ChatGPT served as a usable facilitator for researchers engaging in the software development life cycle, from product conceptualization to

XSL•FO
RenderX

feature identification, and user story development to code generation. GenAI technologies facilitated effective communication and understanding within our multidisciplinary team by providing well-described features and supporting the role of a software engineer. Our findings indicate that the ChatGPT-generated output is comprehensive, albeit with occasional ambiguities that required clarification or adjustment by the research team. The ChatGPT-generated output exhibited a high level of accuracy in capturing the intended requirements. We found that ChatGPT supported a highly efficient development process, producing over 5 days what initially required more than 200 human hours from content and technical experts. The results suggested that by efficiently prompting ChatGPT and leveraging the expertise of our team, we could have significantly reduced the time we invested in initial system modeling and conceptualization phases as well as technical phases of software development (coding). Overall, GenAI technologies like ChatGPT offer a promising approach to efficient software development.

While promising, some significant limitations to ChatGPT's outputs should be noted. In the design phase, while ChatGPT was able to provide general guidance in tool design (eg, app vs web-based vs EHR solution) it was unable to provide evidence to support its rationale for these choices. This lack of reference support has been well-documented and has a material impact on researchers looking to build upon an evidence base for their health technology interventions. Similarly, when asked to provide theoretical frameworks to support behavior change, it offered only a partial list, initially excluding the COM-B (capability, opportunity, motivation, behavior) model upon which the original PAMS intervention was based, and needed prompting from our behavior change expert to provide more specific guidance. In the context of code generation, we focused on testing a specific function, namely the Calculating Patient Engagement feature, which is the core functionality of our software. Initially, we tasked ChatGPT with generating a function to compute a 3-week patient engagement trend. However, the initially generated code deviated from the intended objective and instead calculated a weekly engagement score. Through subsequent iterations, we were able to obtain the desired code. However, the initial attempts exhibited nonidiomatic constructs and contained bugs (no efficient loops and wrong logic). Finally, we observed that ChatGPT overlooked certain suggested features during the design phase, resulting in the generated code occasionally demonstrating unnecessary complexity and disregarding some of the best practices and features of the target programming language. We believe that further iterations would have improved the code quality, encompassing better adherence to coding standards and the inclusion of desired business features, such as handling edge cases and capturing more nuanced engagement trends. Nevertheless, we reached a point of diminishing returns with ChatGPT where we determined that engaging an experienced developer would have expedited the code generation process and ensured a more robust implementation.

These limitations highlight the ongoing importance of human expertise in the development process, especially in scenarios where theoretical expertise, intricate coding practices, and business-specific requirements are involved. The lack of rationale to support the generated results shows the value of having human experts on the team who can interpret the results. ChatGPT needs to be used as a support tool but not the source of truth; thus, we always trusted and relied on human experts to validate the ChatGPT-generated results before moving to the next phase. Overall, it is important to have human experts in the system development process to guide the outputs in terms of reprompting the system (support the decision-making on acceptable output) and ensuring their accuracy. Moreover, results are highly dependent on the quality of the prompts which emphasizes the role of prompt engineering. The results show that well-structured prompts (role + problem description + ask) that infuse human expertise into every iteration are key to obtaining good results (Figure 1). As part of our prompt framework described in the methodology section, results showed that detailed problem explanations, clear asks, and roleplaying are an excellent combination to guide accurate results. We suggest asking ChatGPT questions using different roles, asking for clarification if needed, and in cases of wrong outputs, redirecting the prompts.

Related Work

There is near-universal interest in understanding the impacts of GenAI and large language models (LLMs) on human social structures, including the experience of work and the production of work-related outputs in health care and more broadly [15,16]. In health care, LLMs are poised to impact everything from care delivery experience, diagnostic reasoning and cognitive skills, training and education, and the overall composition of the workforce [17]. These theoretical disruptions are tempered, however, by acknowledging that in its current state, GenAI tools remain suboptimal, with ongoing issues in accuracy, reliability, usability, cost, equity, and ethics.

In commercial spaces, ChatGPT-enabled products designed to assist with coding and software development are already being developed (eg, OpenAI Codex [OpenAI] and CodeGPT [CodeGPT]). These tools can help generate novel code, debug and analyze code issues, assist in code refactoring, and provide code documentation. As yet, however, their usefulness in terms of quality has not been extensively evaluated, and costs and other considerations may make them inaccessible to health care researchers. ChatGPT-enabled tools for front-end design (eg, integrating ChatGPT with Figma [Figma, Inc]), user testing (including synthetic user testing), and prototyping have also been created, all allowing health technology research teams with limited design resources to take advantage of tools from product and experience design to create their interventions. Overall, commercial LLMs have been demonstrated to improve worker efficiency and productivity, through "co-pilot" support services that automate low-skills tasks, organize and present information, and surface insights [18]. Brynjolfsson et al [18] found that a ChatGPT-supported tool providing conversational guidance for customer support agents increased worker productivity by almost 14%. The authors further found that these productivity benefits accrued disproportionately to less-experienced and lower-skill workers, allowing less-skilled or newer workers to experience more rapid gains; the authors posit that high-skill workers may have less to gain from artificial

XSL•FO
RenderX

intelligence assistance due to tacit knowledge reinforcement rather than new knowledge or skill development. Our work suggests that both less-experienced, lower-skill workers and high-skill workers can benefit, with novices benefitting more from new knowledge (if accurate) and skill development and experts benefiting from knowledge validation and offloading of high-effort low-value tasks.

In the academic computer science literature, ChatGPT has been evaluated as a tool for collaborative software design [4], including to improve code quality refactoring, requirements elicitation, and general design solutions [5], and fix programming bugs [19]. Similar findings are reflected in our work, including the caveats of requiring human oversight. Other authors have identified important ethical issues in using GenAI solutions for software engineering, which were not considered in this study [20].

Within health care, a growing body of research has explored the feasibility of GenAI tools (mostly ChatGPT) in a variety of use cases, including answering patient questions [3,21], creating suggestions to optimize clinical decision support [22], generating a history of present illness summaries [23], and overall examination performance [24]. In general, these papers find promising signals for the accurate and acceptable use of GenAI tools, but with many current-state caveats for their optimal, safe, and scaled use. Key areas of concern include reliability (particularly around hallucinations and citation fabrication), reproducibility, and recency of data inputs. While research in this area will continue to grow, as more test cases comparing GenAI performance to that of clinical staff will be undertaken, further work is needed to create validated and generalizable outcome measures. Future work must also ensure that the variety of GenAI tools (including general commercial LLMs, health care–specific LLMs, and internally developed tools) are equally evaluated.

Limitations

There are several limitations to this study. First, no research team members have expertise in prompt generation for GenAI tools; as a result, our prompting reflects the a priori perspectives, biases, and knowledge gaps of our team, and are therefore particularly subject to issues of framing, recall, and confirmation bias that may influence the interpretation of the results. Second, our research team members, who acted as prompt engineers in this study, were highly familiar with the project and participated in the human-based design process; thus, they were aware of what deviations from human-based design to address by reprompting the system. As a result, we have introduced bias in the prompting process and results reflect higher accuracy. Third, the absence of robust tools to objectively measure the "quality" of current ChatGPT outputs poses challenges to accurately and objectively assess its performance. Furthermore, in this case, the output reviewers were not blinded to the human vs ChatGPT outputs, given the complexity of this study and the difficulty in providing enough research context to support independent blind review. Finally, broader limitations of the technology, such as potential hallucinations and concerns about behavioral changes of responses over time, deserve acknowledgment, as they could have implications for the practical applications and long-term viability of GenAI in health care research contexts. Future research efforts should address these limitations to enhance and replicate our findings.

Implications and Future Directions for Exploration

We are considering several future directions for the use of ChatGPT in our digital health intervention development. We envision increasing our expertise in prompt engineering (add expert prompt engineers to the team) to actively use ChatGPT to further develop PAMS features, particularly for additional messaging content. We anticipate this will save our research team considerable time and effort. We may also use ChatGPT to facilitate more time-consuming aspects of our research documentation, including both coding documentation and larger research archival work (eg, meeting minutes and recording intervention decision-making). Overall, we feel ChatGPT and related tools can be effectively leveraged within health care technology research teams with a spectrum of technical expertise, serving to both augment existing skills and supplement skill gaps. For those with expertise in computer science or programming, we imagine ChatGPT can assist by automating high-effort, low-impact tasks or repetitive work that is considered important but often deprioritized as more urgent tasks arise (eg, code documentation). For those without preexisting programming skills, we imagine ChatGPT can offer technical support, including educational tools and skill-building opportunities. Overall, this process will both validate existing knowledge and create new knowledge for teams, as well as potentially improve interteam communication and collaboration.

Conclusions

In this study, we explored the use of the GenAI tool ChatGPT to recreate a novel digital behavior change intervention which our research team had previously developed to support patient engagement and adherence to a commercial dDPP. Specifically, we reviewed and evaluated the capacity and limitations of ChatGPT to support digital health research intervention ideation, design, and software development, finding it a feasible and potential time- and resource-saving tool to support research teams in developing novel digital health products and technologies. At the same time, we identified gaps in ChatGPT outputs that may limit its effective use for both novel and advanced technology developers, particularly around the completeness of outputs. Future directions will include the development of more targeted artificial intelligence–based tools to support health care researchers with all levels of software or engineering skills, as well as the development of improved tools to objectively evaluate GenAI outputs.

Acknowledgments

The National Institute of Diabetes and Digestive and Kidney Diseases (1R18DK118545-01A1, principal investigator: DM) funded this research.

XSL•FO
RenderX

Conflicts of Interest

None declared.

References

1. Risling TL, Risling DE. Advancing nursing participation in user-centred design. J Res Nurs. 2020;25(3):226-238. [doi: 10.1177/1744987120913590] [Medline: 34394630]
2. Dave T, Athaluri SA, Singh S. ChatGPT in medicine: an overview of its applications, advantages, limitations, future prospects, and ethical considerations. Front Artif Intell. 2023;6:1169595. [FREE Full text] [doi: 10.3389/frai.2023.1169595] [Medline: 37215063]
3. Liu S, Wright AP, Patterson BL, Wanderer JP, Turer RW, Nelson SD, et al. Using AI-generated suggestions from ChatGPT to optimize clinical decision support. J Am Med Inform Assoc. 2023;30(7):1237-1245. [FREE Full text] [doi: 10.1093/jamia/ocad072] [Medline: 37087108]
4. Ahmad A, Waseem M, Liang P, Fahmideh M, Aktar MS, Mikkonen T. Towards human-bot collaborative software architecting with ChatGPT. 2023 Presented at: EASE '23: Proceedings of the 27th International Conference on Evaluation and Assessment in Software Engineering; June 14-16, 2023, 2023;279-285; Oulu, Finland. [doi: 10.1145/3593434.3593468]
5. White J, Hays S, Fu Q, Spencer-Smith J, Schmidt DC. ChatGPT prompt patterns for improving code quality, refactoring, requirements elicitation, and software design. ArXiv. Preprint posted online on March 11 2023. [FREE Full text] [doi: 10.48550/arXiv.2303.07839]
6. Rodriguez DV, Lawrence K, Luu S, Yu JL, Feldthouse DM, Gonzalez J, et al. Development of a computer-aided text message platform for user engagement with a digital diabetes prevention program: a case study. J Am Med Inform Assoc. 2021;29(1):155-162. [FREE Full text] [doi: 10.1093/jamia/ocab206] [Medline: 34664647]
7. Lawrence K, Rodriguez DV, Feldthouse DM, Shelley D, Yu JL, Belli HM, et al. Effectiveness of an integrated engagement support system to facilitate patient use of digital diabetes prevention programs: protocol for a randomized controlled trial. JMIR Res Protoc. 2021;10(2):e26750. [FREE Full text] [doi: 10.2196/26750] [Medline: 33560240]
8. Liu V, Chilton LB. Design guidelines for prompt engineering text-to-image generative models. 2022 Presented at: CHI '22: Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems; April 29-May 5, 2022, 2022;1-23; New Orleans, LA. [doi: 10.1145/3491102.3501825]
9. Zhou Y, Muresanu AI, Han Z, Paster K, Pitis S, Chan H, et al. Large language models are human-level prompt engineers. ArXiv. Preprint posted online on November 03 2022. [FREE Full text]
10. White J, Fu Q, Hays S, Sandborn M, Olea C, Gilbert H, et al. A prompt pattern catalog to enhance prompt engineering with chatGPT. ArXiv. Preprint posted online on February 21 2023. [FREE Full text] [doi: 10.48550/arXiv.2302.11382]
11. Wang J, Liu Z, Zhao L, Wu Z, Ma C, Yu S, et al. Review of large vision models and visual prompt engineering. Meta-Radiology. 2023;1(3):100047. [FREE Full text] [doi: 10.1016/j.metrad.2023.100047]
12. Nadareishvili I, Mitra R, McLarty M, Amundsen M. Microservice Architecture: Aligning Principles, Practices, and Culture. Sebastopol, CA. O'Reilly Media, Inc; 2016.
13. Alshuqayran N, Ali N, Evans R. A systematic mapping study in microservice architecture. 2016 Presented at: 2016 IEEE 9th International Conference on Service-Oriented Computing and Applications (SOCA); November 4-6, 2016, 2016;44-51; Macau, China. [doi: 10.1109/soca.2016.15]
14. Mathew S, Varia J. Overview of Amazon Web Services. Amazon Whitepapers. 2014. URL: https://d1.awsstatic.com/whitepapers/aws-overview.pdf [accessed 2024-02-01]
15. Eloundou T, Manning S, Mishkin P, Rock D. GPTs are GPTs: an early look at the labor market impact potential of large language models. ArXiv. Preprint posted online on March 17 2023. [FREE Full text] [doi: 10.48550/arXiv.2303.10130]
16. The impact of artificial intelligence on the future of workforces in the European Union and the United States of America. The White House. 2022. URL: https://www.whitehouse.gov/wp-content/uploads/2022/12/TTC-EC-CEA-AI-Report-12052022-1.pdf [accessed 2024-01-18]
17. Thirunavukarasu AJ, Ting DSJ, Elangovan K, Gutierrez L, Tan TF, Ting DSW. Large language models in medicine. Nat Med. 2023;29(8):1930-1940. [doi: 10.1038/s41591-023-02448-8] [Medline: 37460753]
18. Brynjolfsson E, Li D, Raymond LR. Generative AI at work. National Bureau of Economic Research. 2023. URL: https://www.nber.org/papers/w31161 [accessed 2024-01-18]
19. Surameery NMS, Shakor MY. Use Chat GPT to solve programming bugs. Int J Inf Technol Comput Eng. 2023;3(1):17-22. [FREE Full text] [doi: 10.55529/ijitc.31.17.22]
20. Akbar MA, Khan AA, Liang P. Ethical aspects of ChatGPT in software engineering research. IEEE Trans Artif Intell. 2023.:1-14. [doi: 10.1109/tai.2023.3318183]
21. Ayers JW, Poliak A, Dredze M, Leas EC, Zhu Z, Kelley JB, et al. Comparing physician and artificial intelligence Chatbot responses to patient questions posted to a public social media forum. JAMA Intern Med. 2023;183(6):589-596. [doi: 10.1001/jamainternmed.2023.1838] [Medline: 37115527]

XSL•FO
RenderX

22. Biswas S, Logan NS, Davies LN, Sheppard AL, Wolffsohn JS. Assessing the utility of ChatGPT as an artificial intelligence-based large language model for information to answer questions on myopia. Ophthalmic Physiol Opt. 2023;43(6):1562-1570. [FREE Full text] [doi: 10.1111/opo.13207] [Medline: 37476960]
23. Nayak A, Alkaitis MS, Nayak K, Nikolov M, Weinfurt KP, Schulman K. Comparison of history of present illness summaries generated by a Chatbot and senior internal medicine residents. JAMA Intern Med. 2023;183(9):1026-1027. [doi: 10.1001/jamainternmed.2023.2561] [Medline: 37459091]
24. Strong E, DiGiammarino A, Weng Y, Kumar A, Hosamani P, Hom J, et al. Chatbot vs medical student performance on free-response clinical reasoning examinations. JAMA Intern Med. 2023;183(9):1028-1030. [doi: 10.1001/jamainternmed.2023.2909] [Medline: 37459090]

Abbreviations

AWS: Amazon Web Services
COM-B: capability, opportunity, motivation, behavior
dDPP: digital diabetes prevention program
GenAI: generative artificial intelligence
LLM: large language model
N/A: not applicable
PAMS: personalized automatic messaging system
REDCap: Research Electronic Data Capture

Edited by A Kushniruk; submitted 18.09.23; peer-reviewed by S Mulvaney, I Oakley-Girvan; comments to author 20.11.23; revised version received 27.11.23; accepted 15.12.23; published 06.03.24

Please cite as:
Rodriguez DV, Lawrence K, Gonzalez J, Brandfield-Harvey B, Xu L, Tasneem S, Levine DL, Mann D
Leveraging Generative AI Tools to Support the Development of Digital Solutions in Health Care Research: Case Study
JMIR Hum Factors 2024;11:e52885
URL: https://humanfactors.jmir.org/2024/1/e52885
doi: 10.2196/52885
PMID: 38446539

©Danissa V Rodriguez, Katharine Lawrence, Javier Gonzalez, Beatrix Brandfield-Harvey, Lynn Xu, Sumaiya Tasneem, Defne L Levine, Devin Mann. Originally published in JMIR Human Factors (https://humanfactors.jmir.org), 06.03.2024. This is an open-access article distributed under the terms of the Creative Commons Attribution License (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work, first published in JMIR Human Factors, is properly cited. The complete bibliographic information, a link to the original publication on https://humanfactors.jmir.org, as well as this copyright and license information must be included.

XSL•FO
RenderX