ChatGPT

Roadmap for a Value-Aligned Digital Habit Tracker

Overview: This application aims to help users align their daily digital activities with a chosen set of values extracted from a text/corpus (for example, a philosophy or moral guide). The system will infer value judgments (e.g., *activity X is good/bad or to what degree*) from the corpus, monitor the user's digital life (apps, websites, and possibly journal entries), evaluate those activities against the user's values, and provide rich feedback (visual and textual) on how well the user's habits align with their desired values. Below is a detailed roadmap breaking down this plan into phases:

Phase 1: Corpus Analysis and Value Model Construction

- **Corpus Ingestion & Preprocessing:** Begin by feeding the chosen text or corpus into an NLP pipeline. Clean the text and split it into manageable units (sentences, paragraphs). This corpus is assumed to reflect the user's ideal values or philosophy. For example, it may implicitly or explicitly state what behaviors are desirable or undesirable.

- **Value Statement Extraction:** Use natural language processing to identify value-laden statements in the text. Look for explicit judgments of the form "*X is Y*" (where X is an action/behavior and Y is an evaluation such as good, bad, virtuous, harmful, etc.). Not all value judgments will be so explicit; some may be implied through sentiment or tone. Techniques like dependency parsing or keyword matching (for words like *should, good, bad, virtue, vice, desirable, blameworthy*, etc.) can help extract these statements. The result should be a list or knowledge base of activities (or concepts) and their associated moral/qualitative evaluations as per the text.

- **Representing Values in a Computable Form:** Transform the extracted values into a computational model. If the corpus provides a simple good/bad polarity, you might assign a scalar score (e.g., +1 for "good/wholesome", -1 for "bad/harmful", with 0 for neutral or not mentioned). For more nuanced texts, you may need a multi-dimensional vector to represent different value dimensions. For example, a philosophical text might emphasize several virtues (e.g., *discipline*, *compassion*, *moderation*, *purity*, etc.), each of which could be a dimension in a "values space." An activity X could then be represented as a vector indicating how it scores on each virtue/vice dimension. Recent advances in NLP suggest this is feasible – researchers have shown that moral or value-laden sentiments can be quantified from text by training specialized language models [1]. If feasible, consider fine-tuning a language model on the corpus: given a piece of text, have the model output an evaluation or vector of values. This could capture subtle judgments beyond simple word matching (essentially building a custom "moral evaluator" AI based on the corpus).

- **Semantic Generalization:** The corpus might not mention every possible activity by name, especially modern digital habits. To handle this, implement a way to generalize the corpus's value judgments to new or unseen activities. One approach is to use **word/sentence embeddings**: embed the activities and the corpus statements in the same vector space. If an activity (e.g., "binge-watching YouTube") isn't explicitly in the text, you can measure its semantic similarity to concepts that are discussed. For instance, if the corpus is a Buddhist text that

frequently praises “mindfulness” and warns against “distraction,” and our embedding finds “mindless web browsing” is close in meaning to “distraction,” we can infer a negative evaluation. This vector-based approach could yield a score by seeing how close X is to known “good” concepts vs “bad” concepts in the embedding space. If creating a custom embedding from the corpus is too complex, an alternative is using a general pretrained model (like Sentence-BERT) and guiding it with the corpus content.

- **Augmenting with External Knowledge (if needed):** If the corpus is very sparse or overly specific, it might miss some activities entirely. In such cases, you could incorporate external sources to infer values for an activity. One idea mentioned is using Reddit or other forums to gauge sentiment about an activity. For example, searching for discussions on “Is [X] considered bad/good?” might reveal common opinions. However, use this carefully – **the priority should be the user’s chosen corpus/values**, not general internet opinion, unless the user wants a broader social perspective. A safer augmentation might be using a larger moral lexicon or knowledge base (for example, Moral Foundations dictionaries or other ethical AI resources) to fill gaps. In summary, Phase 1 delivers a **Values Model**: a function or lookup that can take an activity (described in words) and output an evaluation (good/bad or a multi-dimensional value score) according to the spirit of the input text.

Phase 2: User Data Collection – Tracking Digital Activities

- **Activity Logging System:** Set up a system to continuously and passively log the user’s digital activities (initially on desktop, with potential to extend to mobile later). This involves capturing which application is in use and, if it’s a web browser, the URL or page title being viewed. You already have a basic functionality for recording app usage at intervals; build on that. Ensure the logger records timestamps and perhaps durations (how long an app or website was active). This will create a timeline of the user’s activities (e.g., 7:30pm – 8:00pm: YouTube, 8:00pm – 8:30pm: reading PDF named “project.pdf” in a PDF viewer, etc.). According to lifelogging research, modern devices can indeed capture a wealth of such data (applications used, web pages visited, etc.) as part of a personal digital life log [2]. It’s important to store this data securely (preferably locally) to protect user privacy.

- **Data Enrichment:** Simply having raw app names or URLs might not be enough to determine the nature of the activity. Plan to enrich the logs with more semantic labels. For example, if the URL is `facebook.com`, label it as “Social Media”; if the window title contains “YouTube – [video name]”, label it “Watching YouTube (Entertainment)”. This could be done via simple rules or an external API that classifies URLs/apps into categories. Over time, you might build a dictionary of common apps and what type of activity they represent (work, leisure, education, etc.). The more context you can add, the easier it will be to map to the value model.

- **User Journaling Integration:** In case the automated activity log is sparse or misses context (which it inevitably will for offline or nuanced activities), provide an interface for the user to journal or tag their activities. For example, at day’s end, the user could write a short diary entry (“Today I went for a long walk and listened to an audiobook, but I also spent 2 hours scrolling social media.”). This journal text can be analyzed with sentiment or classification (or even just parsed by the system) to add more detail to the day’s activity record. Journaling not only helps capture non-digital activities, but also how the user *felt* about them, which can validate the value alignment (if the user feels guilty about an activity, it likely conflicts with their values). There’s evidence that correlating activities with mood or feelings can yield insights – for instance, one personal data study found activities like “napping” or “staying in during COVID lockdown”

corresponded to low mood scores, whereas spending time with “friends” or “hobbies” led to high mood scores [3]. In our case, the “mood” is a proxy for value alignment; if the user logs feeling positive or proud about something, it likely aligns with their values (and vice versa).

- **Time Correlations and Patterns:** With timestamped activity data, include functionality to analyze temporal patterns. Automatically compute statistics like: *What times of day does the user engage in mostly value-aligned activities vs. value-misaligned?* Are there *autoregressive patterns* (e.g., doing activity X at time T makes doing Y more likely at time T+1)? For example, the system might observe “On days when you start with meditation (value-positive), you tend to have fewer ‘doomscrolling’ sessions later in the day.” Or “Late-night computer use correlates with activities that your values deem unproductive, indicating a willpower drop at day’s end.” These insights can be derived from the logged data (using correlation or sequence analysis algorithms) and will feed into the feedback phase. Essentially, this sets the stage for personalized pattern recognition, so the feedback isn’t just a static score but also highlights *when* and *under what conditions* the user is more or less in line with their values.

Phase 3: Activity-Value Evaluation Engine

- **Matching Activities to Value Judgments:** Now, build the core engine that takes a logged activity (from Phase 2) and evaluates it using the values model (from Phase 1). Start with direct matches: if the activity or its category is explicitly mentioned in the corpus-derived knowledge base, use that evaluation. For instance, if the corpus says “Wasting time on frivolous entertainment is bad” and the user spent an hour on a gaming app, that’s a straightforward match to a “bad” judgment.

- **Semantic Inference for Unseen Activities:** For activities not explicitly covered in the text, implement the semantic generalization approach. For each activity record, create a textual description that can be fed into the model. For example, if the log says `reddit.com/r/news` for 30 minutes, describe it as “reading news on Reddit.” Feed this description to the NLP model or embedding system developed in Phase 1. The model should output an evaluation: perhaps a score on each value dimension, or a single score if we’re using a one-dimensional good↔bad scale. If using an embedding similarity method, you might calculate similarity of “reading news on Reddit” to concepts in the corpus (maybe the corpus praises “staying informed” – positive – but also warns against “idle internet browsing” – negative; the result might be a mixed evaluation that depends on context).

- **Using External Sentiment (Optional):** If the above methods yield no clear verdict, consider a secondary lookup. The idea of searching Reddit (or generally the internet) can be realized by querying for the activity in question coupled with keywords like “good or bad” or related terms. For example, if the activity is “playing video games,” searching for opinions on “playing video games productivity” might show that many consider it a distraction. However, **tread carefully with this approach**. External opinions might not align with the user’s chosen values – they could even contradict them. Ideally, any external data is filtered through the lens of the user’s value system. A refined approach could be to search within communities that share the same philosophy as the corpus (e.g., if the corpus is Stoic philosophy, search Stoicism forums for mentions of the activity). This can provide anecdotal context on how those values view the activity. Still, for the initial version, it may be wise to rely more on the AI model/embedding approach internally, to maintain consistency.

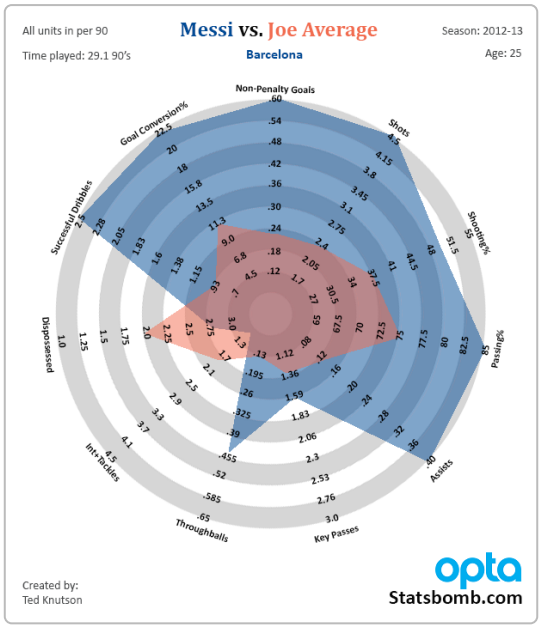
- **Quantifying the Evaluation:** Decide on how to quantify the output for feedback. If using multiple value dimensions, you will have a vector like **V(activity)** = [v1, v2, ..., vN] where each vi is, say, a score from 0 to 100 representing alignment with a particular virtue or value (or possibly a deviation from a vice). If using a single dimension, it could be a score like +10 (very positive/good), 0 (neutral), -10 (very negative/bad), etc. Normalizing these scores is important for comparison. You might derive these by scaling raw model outputs or by calibrating against examples. For instance, if *"studying"* is highly praised in the text, set that as +10 benchmark, and if *"gossiping"* is strongly condemned, that's -10, with others falling in between. This numeric representation will feed into visualizations.

- **Handling Ambiguity and Context:** Some activities might be context-dependent. The same action could be good in moderation but bad in excess (e.g., "resting" is good for recovery but excessive idleness could be bad). To handle this, incorporate context from the logs: frequency and duration. If the user spent 10 minutes on social media, maybe that's not too bad, but 3 hours might be flagged as problematic. You can implement simple rules or thresholds for such cases, possibly informed by the corpus if it mentions balance/moderation. Additionally, if the user journal indicates their mood or self-judgment ("I feel bad I procrastinated"), the engine could weight that in the evaluation (essentially, user's own judgment corroborates the value model's judgment).

- **Test the Evaluation Engine:** At this stage, it's wise to test the mapping on some sample data. Take a few hypothetical daily logs and run them through the engine to see if the evaluations make intuitive sense, especially to the target user. For example, if the user's chosen text is a Buddhist manual, test that activities like "meditation" come out strongly positive, "harmful speech on social media" comes out negative, etc. This will help fine-tune the parameters or rules in the model before showing anything to the end user.

Phase 4: Feedback Interface and Visualization

- **Designing Visual Feedback:** Present the evaluation results to the user in a clear and engaging way. A multi-faceted visual approach is ideal, so the user can quickly grasp their alignment across different values and over time. One proposed method is using a **radar chart (spider chart)** to show the user's "profile" across all value dimensions. Each axis of the radar chart would represent one key value or virtue from the corpus, and the plotted shape shows the user's score in that dimension (e.g., self-discipline, compassion, etc.). Radar charts are commonly used in sports analytics to compare player attributes – they allow visualization of many stats at once in an intuitive shape [4] [5]. For example, the chart below compares a top football player's stats to an average player, using a polygon shape to depict strengths and weaknesses on multiple metrics



. In our case, the "metrics" would be the user's adherence to each value. A quick glance might show, for instance, that the user scores high on "Learning" and "Health" but low on "Mindfulness" for this week, indicating where improvement is needed.

- **Time-Series and Trend Visualization:** In addition to the radar snapshot, include charts that show change over time. For instance, a line chart or bar chart per day that illustrates an overall "value alignment score" for that day. You could also break it down by categories: time spent in value-aligned activities vs non-aligned activities each day. This helps the user see progress (or regress) and identify patterns (e.g., *"Fridays are consistently low alignment days for me"*). If you computed correlations or detected triggers in Phase 2, surface those insights visually or with annotations (for example, mark days with certain events, or use icons to indicate "low sleep night" or "traveling" if those correlate with alignment dips).

- **Qualitative Feedback (Narratives/Alerts):** Not all feedback should be numeric or visual; some users benefit from text-based insights or even conversational feedback. Using the values corpus as a stylistic guide, the app could generate short narrative summaries or tips. For example, "*According to your chosen philosophy, moderation is key – yet 70% of your screen time today was entertainment. Consider reducing this to improve your alignment.*" If possible, make the tone align with the corpus (a user who input a Stoic text might get Stoicism-flavored advice, whereas a user who input a Buddhist text might get a gentler, compassionate tone). This can be achieved by templating sentences with quotes or references to the corpus. You can even highlight direct corpus quotes that are relevant: *"Remember the quote: 'Idle gossip clouds the mind' – today 2 hours were spent on social media gossip, which might be why you felt unsatisfied."* Such context-driven feedback makes the experience richer and more personalized.

- **Interactive Exploration:** Provide a way for users to drill down. The dashboard might allow the user to click on a particular day or a particular value dimension to see more details. For example, clicking on "Mindfulness" could list all the activities that affected the Mindfulness score, along with their individual contributions. Or clicking on a day could show a timeline of that day's activities with color-coding (green for positive, red for negative). This gives users agency to explore and understand, rather than just being told a score.

- **Alerts and Habit Formation Aids:** As an extension, the app could use the data to proactively assist habit change. For instance, if it detects a pattern like "Weekday evenings are your downfall for wasting time," it could suggest or send an alert at that time: *"It's 8 PM – in the past, this is when you often deviate from your goals. Perhaps plan a constructive activity now?"* This moves beyond reflection into intervention. For now, this is a nice-to-have; core feedback is the focus.

- **User Customization:** Ensure the user can adjust how feedback is presented. Some might love the radar chart, others may prefer simple text or a pie chart of "productive vs unproductive time." Also, allow the user to set the *voice* of the feedback to an extent. Since you mentioned "in a sound/voice that you prefer," this could imply different styles or even a literal audio voice. Technically, you could incorporate text-to-speech to read the daily summary in a soothing voice or a motivational tone, depending on user preference. At minimum, letting the user choose the form of feedback (e.g., formal report vs. friendly coach vs. philosophical quotes) can increase engagement.

Phase 5: Testing, Iteration, and Expansion

- **Prototype Testing:** Before fully launching, test the complete pipeline end-to-end with a small set of users (or just yourself). Does the system correctly capture activities? Does the values model output reasonable evaluations? Importantly, do users find the feedback **credible and motivating**? This kind of app must avoid coming off as judgmental in an irritating way – the feedback should feel like it's truly derived from the user's own chosen values (an *ally*, not a nag). Collect qualitative feedback from testers on whether the insights resonate. For example, a user might say *"Yes, when it highlighted that I spent 3 hours on YouTube and reminded me of my goal to be mindful, I actually felt it was right – I did stray from what I want to be doing."* On the other hand, if the system mislabels something (maybe it flagged "reading fiction" as negative because the text corpus only praised work and saw leisure as wasteful, but the user actually values some relaxation), note that. You may need to allow some personalization or tweaking of the value model to account for individual differences.

- **Refinement:** Use testing feedback to refine each component. This could include:

- Improving the NLP model (maybe certain phrases were misinterpreted; you could add more training data or rules).
- Adjusting the weight of time spent in evaluations (e.g., maybe short indulgences shouldn't penalize the user too harshly).
- Enhancing the visualizations (make them clearer, or add a comparison to previous week, etc., if testers want that).

- Streamlining the user interface for journaling or for reading the feedback (UX is important so that interacting with the app becomes a habit itself).

- **Privacy and Data Security Check:** By this point, you'll have a system that deals with sensitive personal data (activity logs, possibly private journal entries). Ensure that data is stored encrypted if possible, and clearly communicate to the user how their data is used. If any external service (even a Reddit search) is employed, disclose it or allow opt-out, since some users may not want their activity queries sent externally. Ideally, the core analysis can happen offline on the user's device for privacy. Modern hardware can handle a lot of this (especially if using lightweight models or periodic analysis). Consider making the architecture such that **the user's data and values never leave their device** unless they explicitly choose to sync across devices.

- **Scaling to Mobile and Other Platforms:** Once the desktop prototype is solid, plan the extension to mobile, as mobile usage is a big part of "digital life." Mobile data will include app usage (which apps for how long) and possibly additional sensors (location, physical activity, etc., if relevant to values like health). You might need to develop a mobile app or use phone APIs to get screen time information. This can hugely enrich the dataset (for example, knowing the user spent 2 hours on TikTok on their phone, which wouldn't show up in desktop logs). The values model and evaluation engine would remain the same, just with more input data. Be mindful of battery and performance on mobile; you might do more batch processing (e.g., analyze the data at night or when charging).

- **Future Enhancements:** With more data and user feedback, consider advanced features. For example:

- **Social Comparison or Community (optional):** If the user is willing, they could compare their value alignment scores with friends or a supportive community (e.g., how others living by the same Stoic principles are doing), to gain motivation or tips. This would require careful anonymity and consent, but could be inspiring.
- **Gamification:** Introduce streaks or leveling up for maintaining alignment, to encourage habit formation.
- **Adaptive Learning:** If the system consistently finds certain user behaviors that the model labels as "bad" but the user disagrees (or vice versa), incorporate a feedback mechanism for the user to correct the model. Over time, the model could adapt to the individual's interpretation of their chosen values (since even within, say, a philosophy, people might emphasize different aspects).

- **Plug-in for Browser or OS:** To get finer data, a browser extension could log specific sites visited in detail or even block/remind when visiting sites that are against values during certain hours (if user wants intervention). An OS-level integration might detect things like multitasking or idling. These are deeper technical challenges and should be optional features later on.

- **Evaluation of Efficacy:** Finally, as the app is used over months, measure if it actually helps users change behavior towards their stated values. This could be done through periodic self-assessments in-app. If it's effective, you'll see increasing alignment scores and users reporting satisfaction. If not, learn whether the issue is in detection (maybe the system misclassifies some value) or in user experience (maybe feedback wasn't actionable enough). Continual improvement will make the application truly valuable to users looking to build better habits aligned with their ideals.

Conclusion

Overall, the plan is ambitious but well-founded. It combines **NLP** for value-based understanding, **lifelogging** for activity tracking, and **data visualization** for feedback – all cutting-edge areas. The roadmap above breaks it into manageable phases: first build the value model from the text, then gather user data, create the evaluation logic, and finally focus on user feedback and iteration. Each part has its challenges (e.g., NLP accuracy, data privacy, user engagement), but there's existing research and technology to leverage for each. The idea of capturing a person's *philosophical values from a text and reflecting their daily life against it* is quite novel. If executed well, it could provide users with profound insights into their habits – essentially holding up a mirror that speaks with the voice of their chosen values. By following this roadmap and iterating based on real-world feedback, you can refine the application into a truly helpful personal development tool. Good luck with the implementation, and enjoy the process of bringing this thoughtful idea to life!

Sources:

- Ramezani & Xu (2024). *MoralBERT: A Language Model for Moral Values.* (demonstrates that moral values and sentiments can be automatically gauged from text with modern NLP) [1].
- StatsBomb (2016). *Understanding Football Radars.* (explains radar/spider charts as a way to visualize multiple performance metrics in one shape, an idea we adapt for visualizing value alignment) [4] [5].
- Dang-Nguyen *et al.* (2018). *Challenges and Opportunities in Personal Life Archives.* (describes how digital devices can log user activities like apps used, websites visited, etc., as part of lifelogging) [2].
- Santos (2020). *1000 Days of Mood Tracking: Data Analysis.* (illustrates correlating daily activities with outcomes like mood; analogous to correlating activities with value-alignment) [3].

---

[1] MoralBERT: A Fine-Tuned Language Model for Capturing Moral Values in Social Discussions
https://arxiv.org/html/2403.07678v2

[2] Challenges and Opportunities within Personal Life Archives
https://doras.dcu.ie/22442/1/icmr2018_pp15_camera_ready.pdf

[3] I've Tracked My Mood for Over 1000 Days: A Data Analysis | by Juan De Dios Santos | TDS Archive | Medium
https://medium.com/data-science/ive-tracked-my-mood-for-over-1000-days-a-data-analysis-5b0bda76cbf7

[4] [5] Understanding Football Radars For Mugs and Muggles • Statsbomb Blog Archive
https://blogarchive.statsbomb.com/articles/soccer/understand-football-radars-for-mugs-and-muggles/