# Road to Victory

Biggest time wasters:

Various phone apps (-> clear rules)

Indigestion from heavy, unhealthy meals (-> more control)

Filler activities like games, series and movies (-> better discipline)

Primary Objectives

Grow cloud services and components knowledge

Review, distil and project knowledge (on)to tangible medium

Train programming principles into short-term memory

Maintain and document sustainable growth

Increase overall fitness (not so great, implicit to all other objectives)

Secondary Objectives

Essence and strategy to doing research, doing sprints and presenting MVP

Essence to low-code, AWS cloud services, cloud architectures, …

Bleeding-edge technologies

Lexicon

POC’s in website portfolio

# Research: Creating an Agile structure for this process

Prompt to OpenAI ChatGPT-4

Im about to embark on a learning journey over the next 5 weeks in which I would like to use a very extensive, technically correct and essence-true implementation of Agile project development. Obviously I cannot simulate (or entertain) the need for creating extensive documentation(s) like technical requirements or business models (from business analytics or business intelligence) but i do want to simulate and create kanban's with users stories and epics that move through the most advised column setup (give me this in bullet points, aside from enclosing question scope). I also want to simulate all of the different kinds of meetings, and center them around their original purpose(s). Possibly there's many more to this, that I either forgot or didn't learn in school. That.s where you come in. Generate me a detailed todo-list (whenever possible, with references to web pages or concepts/names) in order for me to create the setup I've talked loosely about before of which I can be 100% sure it would prepare me greatly for any encounter with Agile development in the future.

Answer

Sure, I'd be happy to help with that! Here's a detailed to-do list on how you can simulate an Agile project development environment:

Week 1: Understanding Agile and its principles

Study the Agile Manifesto: The [Agile Manifesto](https://agilemanifesto.org/) contains the core values and principles of Agile methodologies. This is where Agile was first defined, and it remains the definitive guide to what Agile is all about.

Read about the 12 principles of Agile: These [12 principles](https://agilemanifesto.org/principles.html) are a key part of understanding the Agile mindset.

Learn about different Agile frameworks: Familiarize yourself with Agile frameworks such as Scrum, Kanban, Lean, Extreme Programming (XP), and others. Here's an [overview](https://www.atlassian.com/agile/software-development) of some of the most popular ones.

Week 2: Understanding and Creating a Kanban board

Study the Kanban method: Learn the basics of Kanban, including its principles and practices, on the [Kanban Guide](https://kanbanguides.org/).

Set up your Kanban board: Typically, the board consists of columns that represent stages in the work process. A common setup is: Backlog, To-Do, In Progress, Review, and Done.

Create User Stories: User Stories represent small increments of value that a team can deliver. They usually follow a simple format: "As a [type of user], I want [some goal] so that [some reason]."

Create Epics: Epics are larger bodies of work that can be broken down into a number of smaller tasks (i.e., User Stories).

Week 3: Agile Ceremonies/Meetings

Familiarize yourself with the four key Scrum ceremonies:

Sprint Planning: This is where the team plans the work for the coming sprint. You can read more about it [here](https://www.scrum.org/resources/what-is-sprint-planning).

Daily Scrum/Standup: A short meeting where team members sync up on their progress and discuss any blockers. More details can be found [here](https://www.scrum.org/resources/what-is-daily-scrum).

Sprint Review: This is where the team reviews the work completed in the sprint and discusses what to do next. More information [here](https://www.scrum.org/resources/what-is-a-sprint-review).

Sprint Retrospective: The team discusses what worked well and what needs improvement in their process. You can find more details [here](https://www.scrum.org/resources/what-is-a-sprint-retrospective).

Simulate the meetings: You might consider inviting a few friends or colleagues to help you simulate these meetings.

Week 4: Advanced Agile Concepts

Learn about the role of the Product Owner: The Product Owner is responsible for managing the product backlog and ensuring the team is delivering value. You can find more information [here](https://www.scrum.org/resources/what-is-a-product-owner).

Learn about the role of the Scrum Master: The Scrum Master is a servant-leader for the Scrum Team. They help remove obstacles and coach the team in Scrum. More information [here](https://www.scrum.org/resources/what-is-a-scrum-master).

Understand Agile Estimation Techniques: Techniques such as planning poker, T-shirt sizing, and the bucket system are commonly used in Agile environments to estimate the size of work items.

Week 5: Reflection and Next Steps

Reflect on what you've learned: Go over the concepts you've studied, review the simulated project, and think about what you might do differently in a real project.

Explore certification opportunities: Certifications like Certified Scrum Master (CSM), Professional Scrum Master (PSM), and others could be a next step if you want to demonstrate your knowledge of Agile.

Continue learning: Agile is a vast field with many nuances. Consider reading more books on Agile, following blogs, attending webinars, etc. A recommended book to start is "[Scrum: The Art of Doing Twice the Work in Half the Time](https://www.amazon.com/Scrum-Doing-Twice-Work-Half/dp/038534645X)" by Jeff Sutherland.

Good luck with your Agile learning journey!

Prompt to OpenAI ChatGPT-4

Thanks. In the previous question (and then answer) we've reached the consensus of "what" to do, and a little bit of the "how" as well. Now i would like you to look through every bit of information available about "how" to study and master something like we've discussed previously (how to do agile the methodology) in the most efficient way. Compile that into a how-to for me to rely on, indirectly benefiting from a generalization, also with specific instructions from specific techniques, that depended on a sorting that provides the best sources for information about the topic at hand.

Answer

I'm glad you found the previous response helpful. Now let's dive into the "how" of learning Agile methodologies effectively. Here are some steps and techniques to guide your studies.

\*\*1. Leverage Multiple Learning Styles:\*\*

Different people have different learning styles. Try to incorporate various learning methods to maximize your understanding.

- \*\*Visual:\*\* Use visual aids like diagrams, infographics, and videos. For Agile, flow diagrams can be particularly helpful in understanding the workflow in different methodologies.

- \*\*Auditory:\*\* Listen to Agile podcasts or webinars, and try explaining Agile concepts to others. Some popular Agile podcasts include "Scrum Master Toolbox Podcast" and "Deliver It Cast".

- \*\*Read/Write:\*\* Read books and articles, take notes, and write summaries of what you've learned.

- \*\*Kinesthetic:\*\* Practice by doing. Set up a simulated project or use Agile techniques in a real project if possible.

\*\*2. Utilize Online Learning Platforms:\*\*

Online learning platforms offer structured courses that can guide you through the learning process.

- [Coursera](https://www.coursera.org/)

- [Udemy](https://www.udemy.com/)

- [LinkedIn Learning](https://www.linkedin.com/learning/)

- [Pluralsight](https://www.pluralsight.com/)

\*\*3. Active Learning:\*\*

Active learning is a process that involves actively engaging with the material, as opposed to passive learning where information is simply received.

- \*\*Feynman Technique:\*\* Named after physicist Richard Feynman, this technique involves explaining a concept in simple terms as if teaching it to someone else.

- \*\*Spaced Repetition:\*\* This technique involves reviewing information at increasing intervals over time to improve long-term retention.

- \*\*Mind Mapping:\*\* Create a visual representation of the information to help understand the connections and relationships between concepts.

\*\*4. Collaborative Learning:\*\*

Learning in a group can help you gain different perspectives and learn from others' experiences.

- Join Agile communities online such as the [Scrum.org Forum](https://www.scrum.org/forum) and the [Agile Alliance Community](https://www.agilealliance.org/community/).

- Attend local Agile meetups or conferences if possible.

- Practice pair programming or mob programming if you have the opportunity. This involves two or more people working together at the same workstation, which is a common practice in Agile development.

\*\*5. Reflective Practice:\*\*

Take time regularly to reflect on what you've learned and how you can apply it.

- Keep a learning journal to jot down your thoughts, questions, and key takeaways.

- Regularly review your progress and adjust your learning strategy as needed.

\*\*6. Get Certified:\*\*

Consider pursuing a certification in Agile methodologies. It provides a structured learning path and also validates your knowledge.

- Certified Scrum Master (CSM) from Scrum Alliance

- Professional Scrum Master (PSM) from Scrum.org

- Agile Certified Practitioner (ACP) from PMI

Remember, the goal is not to rush through the material but to truly understand and internalize the Agile principles and practices. Happy learning!