

Onboarding Runbook

Book detailing all the steps and goals of Project Teams onboarding

- [Developer Onboarding](#)
- [Product Lead Onboarding](#)
- [Project Timeline](#)
- [Scope of Roles](#)
- [Standardized Tech Stack](#)
- [Project Onboarding Template](#)

Developer Onboarding

1. Introduction

- **Welcome Message**
 - Warm greeting from the team and leadership.
 - Brief overview of the company's mission and values.
- **Purpose of the Runbook**
 - Explain the goals of the onboarding process.
 - Outline what the developer can expect to learn.

2. Company Overview

- **Company History & Mission**
 - A short history of the company.
 - Mission, vision, and core values.
- **Organizational Structure**
 - Overview of departments and key teams.
 - Organizational chart and leadership bios.

3. Team and Role Integration

- **Team Introduction**
 - Overview of the development team.
 - Introductions to key contacts and mentors.
- **Role Responsibilities**
 - Detailed description of the role.
 - Expectations and key performance indicators.
- **Communication Channels**
 - List of internal communication tools (e.g., Slack, email, project management tools).
 - Best practices for team communication.

4. Technical Environment Setup

- **Hardware and Software Requirements**
 - Necessary equipment (laptop, monitors, etc.).
 - Software installations (IDEs, version control systems, etc.).

- **Access Credentials**
 - Steps to obtain access to systems (VPN, internal networks, etc.).
 - How to request necessary permissions.
- **Development Tools and Environment**
 - Overview of the tech stack and development environment.
 - Installation guides and configuration settings.
 - Accessing repositories and codebases (Git, Bitbucket, GitHub, etc.).

5. Project and Codebase Overview

- **Current Projects Overview**
 - Summary of ongoing projects and priorities.
 - How the developer's work fits into the larger picture.
- **Codebase Walkthrough**
 - Architecture overview and key components.
 - Coding standards and guidelines.
 - Repository structure and navigation tips.

6. Development Processes

- **Agile/Scrum Methodology**
 - Explanation of the development process (sprints, stand-ups, retrospectives).
 - Tools used for task tracking (e.g., Jira, Trello).
- **Code Reviews and Quality Assurance**
 - Best practices for submitting and reviewing code.
 - Testing protocols (unit tests, integration tests, etc.).
- **Deployment and CI/CD Pipelines**
 - Overview of the deployment process.
 - Tools used for continuous integration and deployment.

7. Learning Resources and Documentation

- **Internal Documentation**
 - Location and structure of internal wikis or knowledge bases.
 - How to contribute to or update documentation.
- **Training Materials**
 - Recommended courses, tutorials, or workshops.
 - Links to video tutorials or recorded onboarding sessions.
- **External Resources**
 - Industry blogs, forums, and communities relevant to the tech stack.

8. Security and Compliance

- **Security Best Practices**
 - Overview of security policies and protocols.
 - Guidelines for secure coding practices.
- **Compliance and Data Privacy**
 - Information on regulatory requirements.
 - Steps to ensure compliance within the development environment.

9. Support and Feedback

- **Mentorship and Buddy System**
 - Details on assigned mentors or buddies.
 - How to schedule one-on-one sessions.
- **Escalation Paths**
 - Contact information for IT support, HR, and other key departments.
 - Steps to report issues or ask for help.
- **Feedback Mechanisms**
 - How and when to provide feedback on the onboarding process.
 - Channels for ongoing improvement suggestions.

10. Checklist and Timeline

- **Onboarding Checklist**
 - Step-by-step list of tasks to be completed during the first week, month, and quarter.
- **Timeline and Milestones**
 - Expected milestones and progress reviews.
 - Regular check-in schedules with supervisors.

Product Lead Onboarding

1. Introduction

Welcome to the team! We are excited to have you as a product manager for our next project cycle.

- **Purpose of the Runbook**
 - Explain the goals of the onboarding process.
 - Outline the scope of the product manager role.

2. Blueprint Overview

- Blueprint is a student organization created to serve Non-Profit Organizations (NPOs) by providing them with free web or mobile app solutions.

3. Team and Role Responsibilities

- **Organize project team and NPO meetings**
 - Project team meetings may consist of:
 - **Standup** – Quick updates on progress
 - **Task Review** – Evaluate ongoing/completed tasks and address any concerns
 - **NPO feedback review** (if applicable)
 - **Task Assignment** – Define and assign tasks for the upcoming week
 - **Open Discussion** – Address any additional topics or unplanned updates
 - Weekly meetings will alternate between standup and sprint planning/review. More info linked down below
 - NPO meetings occur monthly and consist of:
 - **Progress Report** – Display project progress to stakeholders
 - **Open Discussion** – Ask any relevant questions regarding the project or project status
- **Define Tasks/Features to be implemented**
 - Create user stories and task descriptions for developers to implement
 - Work with the Tech Lead decide technical implementation of tasks/features
- **Set goals and Timeline**
 - Set goals for each meeting and sprint
 - Determine the timeline of the project
 - What features can be done by the MVP, by the final product handoff?

- What features need to be cut?
 - If so, how should I tell the NPO?
- **Team Dynamics/Hierarchy**
 - The project team hierarchy is as such: PL --> Lead Developer --> Developers
 - PLs have the final say in all decisions
 - Decide what can and cannot be done
 - Determine priority of tasks/features
- **Track Project Progress**
 - Utilize time management applications to track status of project
 - Github taskboard, Jira, Notion, etc...

4. Learning Resources and Documentation

- For Scrum/Agile Methodology
 - <https://wiki.sitblueprint.com/books/onboarding/page/product-management-agile-at-blueprint>

5. Communication Channels

- **For Project Team**
 - GitHub Task Board
 - Meeting Minutes
 - Discord Channels
- **For NPO**
 - Google meet, zoom, etc...
 - Email
 - For fast/urgent communication
 - Slack, Discord, etc...

Project Timeline

Summer

End of July

- **Project Proposal Discussion**

August

Pre-Semester Preparations

- **Meeting with NPO**
- **Confirm project for the year**

Communication & Collaboration Setup

- **Clarify communication channels:**
 - Options: Slack, Discord, Email (choose one that is easily accessible and ideally faster than email)

Project Kickoff & Design

- **First Project Team Meeting**
- **Start Design Mockups**
- **Develop Software Design Documents:**
 - Architecture documents
 - Database schema
 - Tech stack decisions
 - API interface specifications
- **NPO Involvement:**
 - NPO reviews and provides feedback on designs prior to semester start
 - This process should involve the entire team

Team Responsibilities

- **Tech Lead / VP Tech Responsibilities:**
 - Setup GitHub repository
 - Setup pipelines
 - Establish staging environment
- **Project Manager (PM):**

- Begin writing master issues for general features, following scrum methodology and user stories
 - **Technical Aspect:**
 - TECH LEAD to write technical features of issues
-

Fall Semester

September

- **Monthly Meetings:**
 - Meeting with NPOs for ongoing feedback

October

Midpoint Review with VP Projects and Tech Leads:

- **Feedback Focus:**
 - Input from NPO and team members
 - Improvements in operations
 - Project updates
 - General team feedback
 - Team member statuses
 - Efficient practices
 - Necessary operational changes
 - Design team review ensuring alignment with project proposal

November - December

Semester Review with VP Projects and Tech Leads:

- **Review Points:**
 - MVP progress
 - Next steps (if necessary)
 - Project updates
 - General team feedback
 - Team member statuses
 - Efficient practices
 - Operational changes
 - Design team review for cross-checking with project proposal

End of Semester

- **NPO Meeting:**

- MVP Showcase
 - Winter Break begins
-

Winter Break and Spring Semester

Transition: Winter Break to Beginning of Spring Semester

- **Project Team Adjustments:**
 - Possible swaps between Developers and PMs
- **Spring Onboarding Meeting:**
 - Involvement of Tech Leads, PMs, and VP Projects, conducted BEFORE the semester starts
 - Implement operational changes
 - Restart NPO communication
 - Confirm project readiness

Early March (Before Spring Break)

- **Midpoint Review with VP Projects and Tech Leads:**
 - **Review Areas:**
 - Feedback from NPO and team members
 - Improvements in operations
 - Project updates
 - General team feedback
 - Team member statuses
 - Efficient practices
 - Operational changes
 - Design team review confirming alignment with project proposal

During/After Spring Break

- **Operational Changes:**
 - Plan and implement further operational improvements

End of April

- **Final Product Demo:**
 - Presentation for VP Tech, VP Projects, and other stakeholders
 - Testing and final adjustments
 - Final cross-check with project proposal

May

- Final Product Handoff

Scope of Roles

This document describes the overall responsibilities of each role on a project team. For more detail on each role, please see their respective runbooks. Anyone participating in a project team has discretion to comment on any features including frontend and backend architecture and implementation, UI/UX designs, and any other internal discussions.

Tech Lead Role

- **Project Scoping:** Meet with the non-profit organization to understand their project needs and determine the appropriate tech stack, given their requirements and your team's abilities. The VP Projects and your Product Manager will be available to support you throughout this stage. You will also be expected to meet consistently with the NPO to ensure that you are meeting their requirements.
- **Project Planning:** The goal is to implement an Agile (Scrum) project methodology that emphasizes iterative development, frequent communication, and team collaboration to deliver high-quality products. Setting bi-weekly goals for your developers and dividing features into manageable tasks will help enable consistent progress.
- **Building Team Culture:** Blueprint and its teams should be seen as more than just a "job". While the work may be challenging, we want this to be an enjoyable experience for all of those involved. The Exec Team will host GBMs and events throughout the semester to help build the culture within Blueprint as a whole. While most of your team's work may be done asynchronously, implementing practices such as pair programming, or hosting live in-person work sessions can prevent your team from feeling isolated.

Tech Lead Responsibilities

- Communication
 - Attend Monthly NPO meetings
 - Walk through feature/progress reports
 - Ask relevant technical questions
 - Project Team Meetings
 - Address any technical issues raised by devs
 - Decide technical implementation of features/tasks
 - Delegate tasks
 - Features
 - Review, comment, and approve PRs
 - Delegate tasks
 - Complete assigned tasks
-

Product Manager Role

- **NPO Communication:** They will be the team's primary channel of communication with the assigned non-profit organization (NPO). You are responsible for ensuring that your team is meeting the NPO's specifications. One important tip is to consider guiding the discussion from the target user's perspective.
- **Project Planning:** The goal is to implement an Agile (Scrum) project methodology. Alongside your Technical Lead, you will assist in the bi-weekly/monthly planning to ensure that your team meets your goals.
- **Building Team Culture:** As much as this is an opportunity to develop something tangible, it is also an opportunity to be part of a community where we can learn from each other! At the end of the day, your team members are students, and we want everyone's experience to be rewarding and fun. While most of your team's work may be done asynchronously, implementing practices such as pair programming, or hosting live in-person work sessions can prevent your team from feeling isolated.
- **(Optional) Developing:** The product manager is a leadership role, so the aforementioned responsibilities would be expected, however you are more than welcome to assist in the developing to help build your own skill set and be more informed about the project.

Product Manager Responsibilities

- Communication
 - Communication Channels
 - Manage Discord project team channel
 - Facilitate communication between NPOs and Project Team
 - Via Email, Slack, Discord, at PMs discretion
 - Manage task organization software(Github Taskboard, Jira, etc...)
 - Meetings
 - Organize and lead Project Team Meetings
 - Create agenda
 - Standup
 - Goals of meeting
 - Task delegation
 - Iterate feedback from NPO
 - Organize NPO Meetings
 - Feature/progress report
 - Ask questions regarding:
 - Features
 - Timeline
 - Scope of Project
- Features
 - Determine priority of features
 - Write Masters feature issues/tasks
 - To be populate by Tech Leads
 - Delegate tasks

- Quality Assurance
 - Test Features for functionality
 - Write according bug/fix tasks that address necessary changes
 - Approve UI/UX and designs
 - Test functionality of buttons, hyperlinks, embedded videos, etc..
-

Developer Role

- **Implementing Features:** The specific technical skills you need will be dependent on your project specs, but as we are expecting to build web-based applications, general frontend and backend skills are necessary. Though projects will come with their unique learning curves, your team is balanced with diverse skills, and your Technical Lead will try their best to ensure you are working on something that you are both comfortable and interested in!
- **Practicing Agile Methodology:** Agile is a project management methodology that emphasizes iterative development, frequent communication, and team collaboration to deliver high-quality products. This is used amongst many tech companies, but no worries if you are unfamiliar with it: we'll be teaching and practicing this with our projects!
- **Documentation:** Not only is it important for yourself and your team to understand the components of your project, but your documentation will be key in the hand-off process to the NPOs.

Developer Responsibilities

- Communication
 - Attend all project team meetings
 - Provide progress report on tasks
 - Ask necessary questions
 - Receive task assignments
 - Features
 - Complete assigned tasks
 - Review, comment, and approve PRs
-

Designer Responsibilities

- **Wireframes and Styling Guides:** You will create wireframes (mock-up layouts of the application requested) and decide styling using any method you're comfortable with. Keep in mind, you will be working closely with the Product Manager to ensure your designs fall under the NPO's interests/branding, but we still encourage you to be creative with it!
- **Content:** You will confirm all the content being displayed on the website with the NPO. (The NPO should handle the bulk of this.) This could range from text, images, and additional graphics.
- **(Optional) Developing:** As a Designer your main responsibilities are what are outlined above. Nonetheless, you are more than welcome to assist in the development to help

build your own skill set and be more informed about the project!

Designer Responsibilities

- Communcation
 - Attend all Project Team meetings
 - Attend necessary NPO meetings(summer, early fall)
- Features
 - Complete UI/UX designs of project
 - Complete task assignment

Standardized Tech Stack

To ensure consistent development quality, maintainability, and smooth onboarding, all Blueprint projects use the following standardized tech Stack.

Frontend Stack

Component	Docs
React	https://react.dev/
TypeScript	https://www.typescriptlang.org/docs/
Vite	https://vitejs.dev/guide/
TailwindCSS	https://tailwindcss.com/docs

Frontend Best Practices

- Use functional components + hooks only
- Use Zod for input validation
- Always type props and state - do not default to any
- Keep styling with Tailwind utility classes; avoid writing custom CSS unless necessary

Backend Stack

Projects use one of:

Option A: Java + Spring Boot

- [Spring Boot](#)
- [Spring Data JPA](#)

Best practices

- Use layered architecture: controller -> service -> repository
- Use DTOs to avoid returning raw database entities
- Prefer Postgres types (UUIDs over integers for IDs)

Project Onboarding Template

[Insert NPO name] Onboarding

NPO	[Insert NPO name, hyperlink to their website]
Primary Contacts	[Insert Tech Lead and PM contact (Name, role, position, email, discord handle)]
Repository Links	[List of repositories used for the project in the following format (repository name with hyperlink to gh repo)]
Issue Tracker	[Direct link to gh issue tracker]

Overview

Provide a short description of the NPO [What do they do? Where are they located? Group of people their services are focused on?]

Provide a high-level description of what the project does, what problem does it solve, who will be using it?

Architecture

Include a short explanation of the system's main components:

- Core Services
- Frameworks. If you are using a niche framework (PayloadCMS, Glom) include a link to their documentation.
- Databases
- Authentication mechanisms

[Insert the architecture document created before project kickoff]

Challenges

Give a prioritized list of all main challenges your team is currently facing and a possible path to solve them. Encourage them to contribute to the problem solving brainstorm.