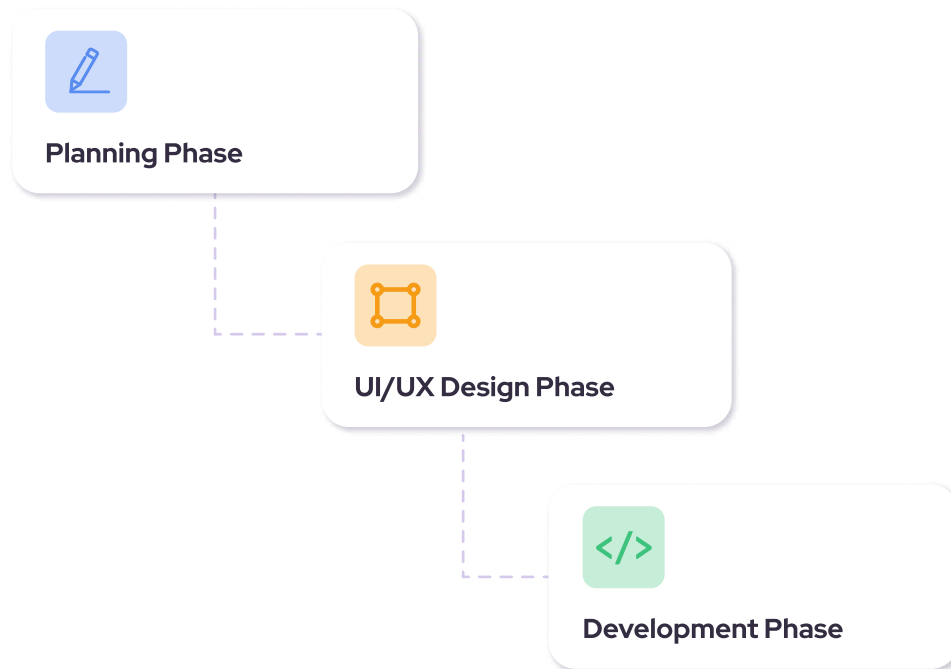


Go-to-Market Playbook



Our go-to-market playbook showcases how we transform your idea, from start to finish, into an exceptional custom software solution. This blueprint details how we plan, design, develop, manage projects. A playbook isn't complete. And never will be. A blueprint is a living document we continuously revise, modify, adapt, and adjust, as time goes on, to meet the market evolution.





Planning Phase

- 6 — Agreement
- 6 — Research
- 7 — Architecture
- 7 — Design
- 10 — Tech Spec
- 11 — Project Estimation



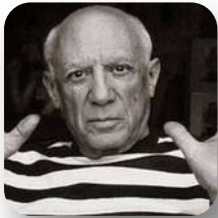
Design Phase

- 17 — Research & Analysis
- 19 — UX Design
- 23 — UI Design
- 26 — Testing, Design Validation & Delivery to Development



Development Phase

- 33 — Active Development
- 45 — Launch
- 46 — Post-Release Support & Maintenance



Pablo Picasso

painter, regarded as one of the most influential artists of the 20th century

”

Our goals can only be reached through a vehicle of a plan, in which we must fervently believe, and upon which we must vigorously act. There is no other route to success.

Why?

Some business owners consider quick briefing a pretty simple formula when creating new products. They often either overlook or skip careful planning in their development model. But it is not as easy as it seems. What's happening when one chooses bad briefs? What's happening is that reality doesn't meet expectations. What's happening is that you face the risk of idles, skips, delays, undefined budget.

The reality is that competent planning is no longer a matter of debate. It is a necessity. The planning phase allows you to check your project—how it will work, how it will look, how long it will take to develop a product, and how much it will cost.

Based on our experience, we have created our approach that ensures getting things done. Our goal is to understand your problem and identify your business targets to develop a forward-looking, business-focused, user-centered, and result-oriented go-to-market plan. So what it's all about?

Planning phase allows you to:



Explore the international market as rapidly as possible



Identify risks in advance and avoid them



Get the 'helicopter view' of the project, its top priorities, and core features



Work without idles, skips, and delays



Receive a detailed tech spec, a long list of features and metrics



Plan scaling, extension, and further releases



Get the visual design of the entire application



Release all stages accurately by timeline



Plan the whole budget up to the cent

How?

- 1 ————— **Agreement**
- 2 ————— **Research**
- 3 ————— **Architecture**
- 4 ————— **Design**
- 5 ————— **Tech Spec**
- 6 ————— **Project Estimation**

Agreement

First of all, we sign an agreement considering intellectual property rights, ownership of an app's source code, design, and content. We define the responsibilities, deliverables, payment structure, workflow, and deadlines. To secure your business idea, we can sign a confidentiality agreement (NDA) as a promise that all data conveyed will be maintained in secrecy.

Research

After that, we analyze your business niche, explore current market trends, forecast the future outlook. We evaluate current and potential competitors' success stories and failures, identify all other relevant aspects. After gathering all the findings and opportunities, we determine your product course—project goals, what is the problem we are to solve, target audience we solve this problem for, timeline, players, and what solutions we are to deliver.

Architecture

Based on your business needs, tech requirements, and other details clarification, we define a project architecture—what functionality and customer solutions to be implemented, what languages, frameworks, libraries, servers, UI/UX solutions, QA and DevOps services to be combined to guarantee fast and easy-to-use final product. We provide you with architecture that is built to last and can ensure your system quality and longevity.

Design

Once the architecture is ready, we start crafting visual solutions. We apply design thinking and user-centric principles to understand, validate, test, and visualize ideas and primary user flows via static wireframes and clickable dynamic prototype. Static wireframes display interactions between screens, the same as they are in the working app. A clickable prototype, in turn, fits the final user interface as far as possible by simulating the real app or website functionality and visual solutions. After reviewing all the wireframes, receiving feedback from you, and specifying the details, we start creating an initial visual treatment.

Agreement

Michael

John



| Search



←] Log Out

Tech Spec

At this phase, we write a tech spec, which allows appropriately thinking things through, avoiding wasting time on dead-end solutions, or building the wrong something.

Tech spec includes a set of actionable use cases and user stories, which describe user interactions that the software must provide. It details the software system to be developed, functional, and non-functional requirements so as performance, operational requirements, constraints to be met, performance risks, business logic. Depending on a project, a blueprint may include such options as big data processing (applying big data algorithms), performing a large number of operations, multilevel computing, scalability, extendibility, application modes (online or offline), parts reuse, B2B interaction mechanisms, multilingual data and/or interfaces, data logging or versioning, interaction with external resources, security issues, applying automated testing, unit tests, UI tests.

Project Estimation

After finalizing a tech spec, we provide you with a project estimation based on the initial product backlog, and answers three crucial questions—what needs to be done, for how long, and at what cost. Project estimation is broken down into high-level elements, workforce, and timeframes and visualized via a Gantt chart, which allows defining each step of development and managing the sprints and epics with high accuracy. Every phase represents a detailed workflow, with the low- and high-range estimates.

Clear goals, clear results. Immersive planning of your project is a solid base for your project success. Efforts and time spent on planning allow eliminating delays, minimizing overhead costs, and delivering on time.

Who Is on Our Team?

Each project is matched with vetted experts, working in close cooperation and capable of solving complex large-scale tasks:

- *CTO*
- *Backend and frontend developers*
- *Business Analyst*
- *UI/UX Designer*
- *Project Manager*
- *Architect*

What's Next?

After we have planned your project, we start crafting design. We tackle creating solutions that sell—from research, wireframes, high fidelity clickable prototype to final release to development.

Optimistic	Likely	Pessimistic	Average
10.50	17.60	27.30	18.66

Optimistic	Likely	Pessimistic	Average
10.50	17.60	27.30	18.66



Thomas Watson Jr.

businessman, second president
of IBM



Good design is good business.

Why?

Design is no longer merely a fad, a wow factor, or just a beautiful picture. Robust design is thoroughly planned, right down to the last detail, to provide personalization, simplicity, convenience, care, accuracy, equal and comfortable experiences for all audience segments.

Design is about personal responsibility. That is why we create a human-centered design combined with inclusive mindset to find real people's pain points to build a real product that solves real needs.

If done correctly, good design sells, since your brand becomes a friend and advisor people are eager to rely on and follow, not just a firm that tries only to sell some services or stuff.

How?

- 1 ——— **Research & Analysis**
- 2 ——— **UX Design**
- 3 ——— **UI Design**
- 4 ——— **Testing, Design
Validation & Delivery
to Development**

Research & Analysis

Meeting with You

First, we hold a meeting with you to ensure that we have a shared vision and a clear idea of the future project—what it must become. Sometimes, clients come to us at different stages of the development process. In this case, we define what has been done already on your project to build a visual roadmap considering features to be implemented.

During a meeting, we will ask you to clarify:

- *How do you see your brand?*
- *How do you want people to hear and feel about it?*
- *What is your message?*
- *What is the concept you want to convey? In what way?*
- *What are the pain points and problems do your end users have?*
- *What solutions must you deliver to solve the problems?*
- *What are the goals you are eager to achieve?*
- *How long should it take?*
- *What are the desired outcomes?*
- *How long should it take?*

Your answers are the basis for us to frame the problems and find specific solutions.

User Research

Your product must solve real problems of real people. That is why, we start our design workflow by figuring out who are your end-users and what they really want. For that, we do these things:

observing user behaviors, needs, and motivations to understand what the user experience is really like and feel what your users feel
identifying a problem space for which we are to propose solutions
establishing business targets and scope of your project
setting up design strategy according to user needs, and project goals defined to convert it into the user flows

Competitive Analysis

Next, we will perform a competitive analysis to define and evaluate similar apps created in the same niche by competitors as well as determine success stories and failures. All that will help us figure out the right visual strategy to follow and identify specific opportunities so that you could step ahead of competitors.

UX Design

We provide UX design workflow in such a way as to ensure an effective and comprehensive solution to your users' problems, create a product that is easy to understand and use, increase brand awareness, build strong relationships with your target audience. To make something so great that keeps users coming back to, again and again. We continually change and modify UX design to meet the fast-paced market requirements.

User Persona

After completing user research and competitive analysis, we outline user persona—a blueprint that showcases who your ideal customer is, and identifies what he or she needs to get from your product. We build a user persona upon a clear problem statement, user behaviors, needs, and motivations.

To discover your user persona, we will ask you to respond to these questions:

- *Who is your ideal end user?*
- *What are behavior patterns of your users?*
- *How and why do user behaviors change?*
- *How can you influence user behavior?*
- *What are the current needs and goals of your target audience?*

- *What issues and pain-points do they face now?*
- *In what way can your product solve these issues and pain-points?*

Name: **Aaron Ackles**

Occupation:
Software Engineer

Status: **Single**

Frustrations:

Housekeeping takes time
that could be spent on
self-improvement and v

**'I want to find proper
repair and cleaning
services in a few taps.'**

Goals:

**Find on-demand repair
and cleaning services.**



Building empathy with your target audience allow us to build a successful strategy, focus efforts, define specific functionality and smart design solutions that perfectly fit your business case.

User Flow

After outlining user persona, we start crafting user flow charts. These visual representations help us demonstrate the core steps end users must take to achieve targets set by mapping out each step. Flow charts help you, as a client, and our team gain a shared understanding of the product as a whole. And that's how:

outlining all the application or website screens and components requirements

defining features to be implemented and how they will work

validating expected outcomes and overall effectiveness to tackle usability issues and create an intuitive interface

iterating faster through ideas

improving communication across the entire team and determining development requirements beforehand

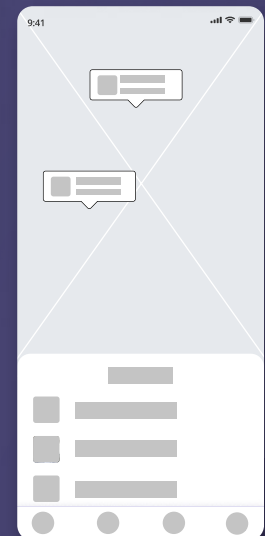
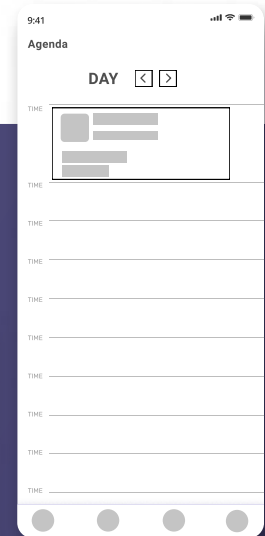
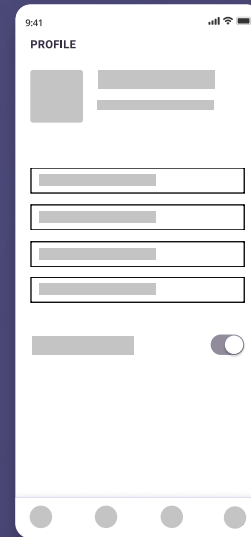
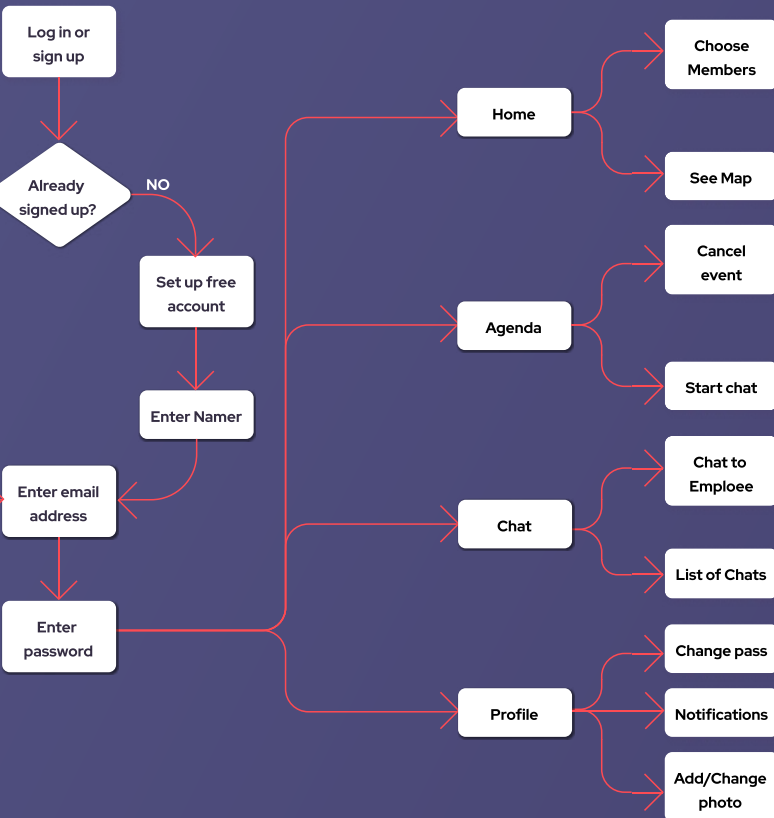
Low-Fidelity Wireframes

At this stage, we create low-fidelity wireframes to primary views of the future app. Wireframes are not about the final design.

End-users will never see them. Wireframes are about platforms that ensure UI/UX designers and coders share the same vision as a client and act as a basis from which experts create a real product.

Low-fidelity wireframes allow:

- Defining visual and typographic hierarchy on user interfaces
- Planning transitions and interactions
- Adjusting the general interface
- Simplifying features representation to developers



UI Design

After finalizing UX wireframing, we move to the user interface to craft style concepts, brand style guide, and mockups.

Style Concepts

Style concepts showcase various variations of the entire app look and feel. We usually offer two or three style concepts, each of which has a unique look and feel, so that you could choose the best suitable one.

Brand Style Guide

Based on the chosen style concept, we start creating a brand style guide. This blueprint details the visual expression of your brand. It combines standards for grid system, layout, color palette, typography, iconography, logos, imagery, accessibility, as well as other visual solutions to be applied to the app.

Our UI/UX design department also works within the existing brand style guide.

If needed, we can create a living brand style guide to combine your brand visual style with the requirements of application development guidelines for iOS or/and Android. This blueprint is synched to the current front-end code.

A brand style guide remains relevant for the years to come, or until you make rebranding.

Typography

H1/ Rubik bold/ 20

H2/ Rubik Bold/ 17

Subtitle/ Rubik Medium/ 17

Body/ Rubik Regular/ 15

Caption/ Rubik Regular/ 10

Colors



Text Main



Text Secondary



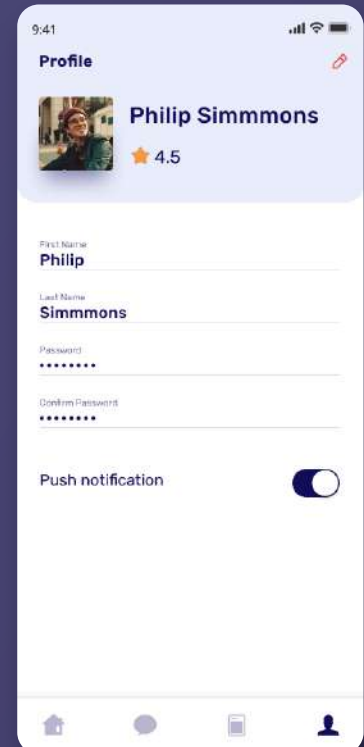
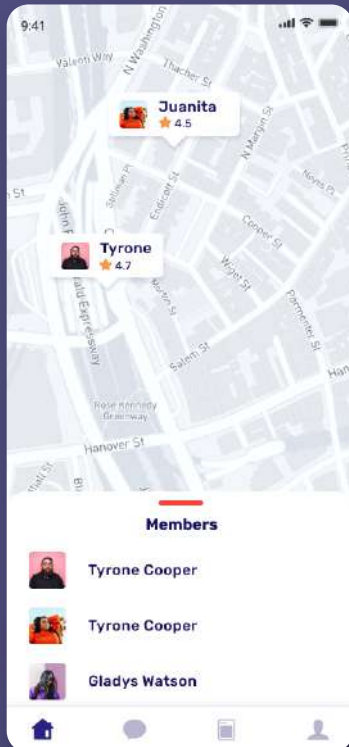
BG



Accent

High-Fidelity Mockups

Now, we create high-fidelity mockups that represent how the app is going to look like after its launch. Mockups help us make final arrangements concerning design elements by combining different options of visual side and choose the one that looks the best. We show the mockups to other team members to get feedback on what changes to make to ensure the best possible result.



Prototype & Estimation

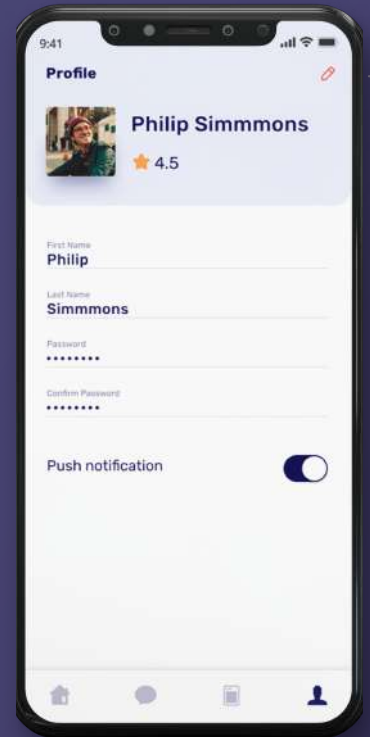
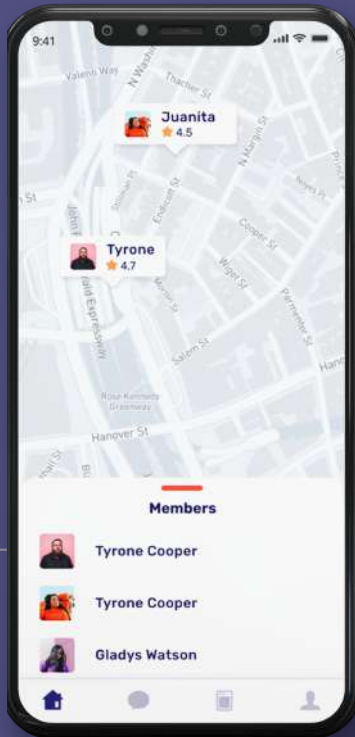
Once mockups are finalized, we start crafting high-fidelity clickable prototype(s) to use it (them) for further development estimation.

High-Fidelity Clickable Prototype(s)

High-fidelity clickable prototypes showcase how the application will function once developed. Prototypes allow testing the app long before its development, as well as gather findings and deliverables that would otherwise be missed. With prototypes, we gain more accurate estimates. And you, as a business owner, receive a powerful tool for getting budgets approved or pitching investors.

Estimation

Clickable prototypes help gain a shared understanding of what is to be built. Based on it, we create an ultimate development estimate—how the app will work, how it will look, how long it will take to develop it, how much it will cost. We outline planned releases with defined timelines and expected costs.



Testing, Design Validation & Delivery to Development

Before the launch, our QA engineers test clickable prototype(s) to identify and correct potential design flaws to ensure everything goes right. After a thorough review, our Q&A team validates the design and gives the green light to delivery to development.

What's Next?

Sending final prototype(s) to development.

Development Phase

```
83
84 <?php if ($useDefList) : ?>
85 <dl class="article-info">
86 <dt class="article-info-term"><?php echo JText::_('COM_CONTENT_ARTICLE_INFO'); ?></dt>
87 <?php endif; ?>
88 <?php if ($params->get('show_parent_category') && $this->item->parent_slug != 'root') : ?>
89 <dd class="parent-category-name">
90 <?php $title = $this->escape($this->item->parent_title);
91 $url = '<a href="' . JRoute::_($ContentHelperRoute::getCategoryRoute($this->item->parent_slug)) . '"' . $title
92 <?php if ($params->get('link_parent_category') AND $this->item->parent_slug) : ?>
93 <?php echo JText::sprintf('COM_CONTENT_PARENT', $url); ?>
94 <?php else : ?>
95 <?php echo JText::sprintf('COM_CONTENT_PARENT', $title); ?>
96 <?php endif; ?>
97 </dd>
98 <?php endif; ?>
99 <?php if ($params->get('show_category')) : ?>
100 <dd class="category-name">
101 <?php $title = $this->escape($this->item->category_title);
102 $url = '<a href="' . JRoute::_($ContentHelperRoute::getCategoryRoute($this->item->catslug)) . '"' . $title
103 <?php if ($params->get('link_category') AND $this->item->catslug) : ?>
104 <?php echo JText::sprintf('COM_CONTENT_CATEGORY', $url); ?>
105 <?php else : ?>
106 <?php echo JText::sprintf('COM_CONTENT_CATEGORY', $title); ?>
107 <?php endif; ?>
108 </dd>
109 <?php endif; ?>
110 <?php if ($params->get('show_create_date')) : ?>
111 <dd class="create">
112 <?php echo JText::sprintf('COM_CONTENT_CREATED_DATE_ON', JHTML::_('date', $this->item->created, JText::_('DATE_FORMAT_
113 </dd>
114 <?php endif; ?>
115 <?php if ($params->get('show_modify_date')) : ?>
116 <dd class="modified">
117 <?php echo JText::sprintf('COM_CONTENT_LAST_UPDATED', JHTML::_('date', $this->item->modified, JText::_('DATE_FORMA
118 </dd>
```

MacBook Pro



Everybody Knows

”

Discipline is the best tool.
Design first, then code.
Don't patch bugs out, rewrite them out.
Don't test bugs out, design them out

Why?

The modern digital market is continuously developing and extending. Apple and Google alone offer about 2 million apps each, so it becomes hard to compete. That's why you, as a progressive business owner, should not only follow the trends and elaborate unique app concepts but follow its progress and adjust to the audience's needs.

Advanced web and mobile solutions guarantee superior, fast, seamless, flexible, and easy-to-use final product.

We help improve and bring your business online through end to end custom software solutions so that you can smoothly run your business everywhere, with no worries, without any barriers.

We create not just apps. We deliver solutions that everyone will always long to use.

How?

- 1 ————— **Active Development**
- 2 ————— **Launch**
- 3 ————— **Post-Release Support
& Maintenance**

Active Development

Establish Milestones

The development process is broken down by milestones. They allow keeping a project on track, managing deliverables, minimizing risks, and keeping within a clear time frame. Each milestone lasts a month and implies specific tasks that our team must complete by the deadline.

Milestones include:

- *fixed beginning and end dates of each project phase*
- *deliverables*
- *approvals*
- *meetings and presentations appointment*
- *force majeure events*

Sprint Planning

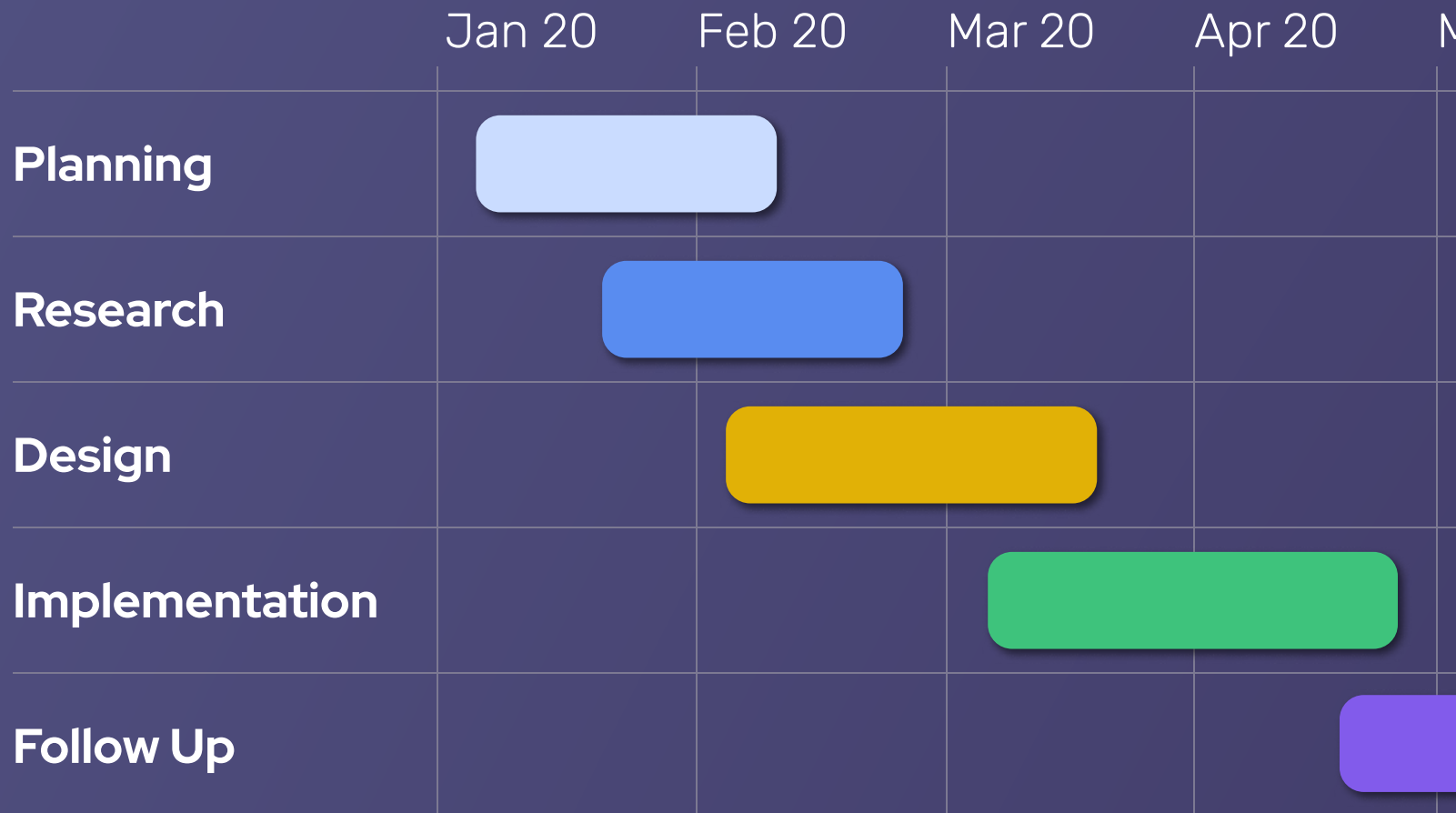
Based on the scheduled milestones, we plan one- or two-week sprints. We determine the sprint goal and the project backlog items we are to work on during the sprint and define how to complete them. For each sprint, developers are assigned a number of tasks from the project backlog with estimated hours. Tasks contain what to build, relevant assets, and criteria.

Sprint Execution

Sprint execution starts after sprint planning and ends with sprint retrospective. For instance, sprint execution for a two-week sprint takes up from eight to ten working days. Sprint execution covers planning tasks, performing and managing tasks, daily stand-up meetings, weekly client meetings, hours reporting, collaboration via appropriate project management software. Our team defines what way to meet the sprint goal will be supreme.

Sprint Demo

After completing a sprint, we provide you with a sprint demo or several ones to ensure that we are on the right track. Sprint demo allows you to be up to speed with the progress of the software development process, provide feedback, as well as outline amendments to the project backlog for planning further sprints.



Automated Testing

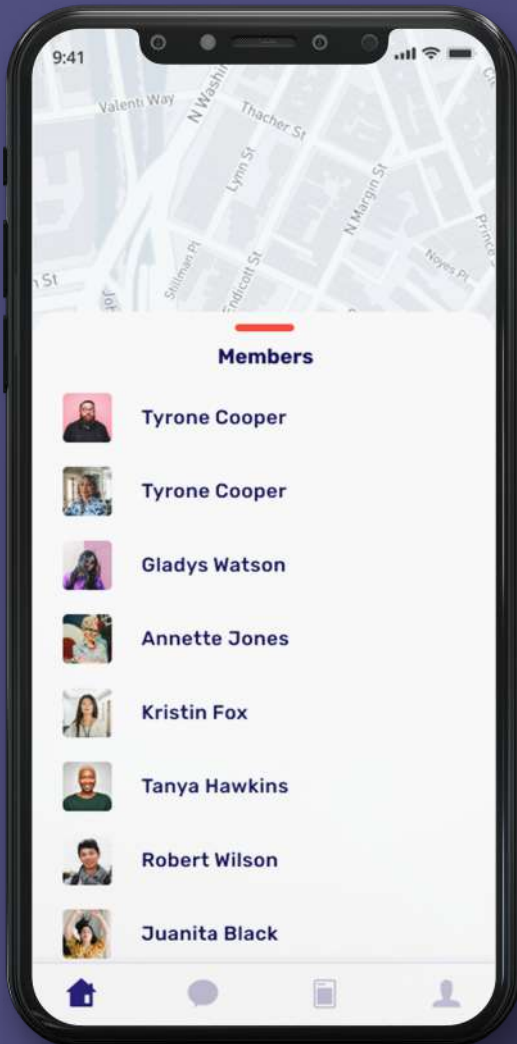
Developers conduct local testing by running automated unit tests, checking code quality, or processing manual checks. Automated testing eliminates the risk of the application performance gap, bugs, errors, freezes, crashes in the long run.

Unit testing

Unit testing is about testing the smallest testable parts of an application to ensure that every unit/component performs precisely as it was designed.

End-to-end testing

End-to-end testing is about testing the app performance and an interface flow, from start to finish, via real user simulation testing, validating component integration, and ensuring successful data integrity.



CI/CD

CI/CD testing methodology performs an integration function, including various types of automated testing at every stage, simplifying merging newly committed code into the main codebase. CI/CD platforms ensure regular automatic build and testing of the current project version to identify defects and solve integration issues quickly.

In a typical, cascade-type, development model, where coders work independently on various parts of the system, the integration stage is the final phase. If errors are detected, it can unpredictably cause unforeseen project delays. Continuous integration allows reducing labor-intensity and making software development workflow more predictable due to early and constant detection and elimination of errors and inconsistencies, minimizing risks for each build and speeding up features launch.

Code Review

We follow GitHub flow to manage branch-based workflow. Each project is divided into specific parts, which are assigned to responsible developers who complete tasks separately. Once a developer writes code, he or she pushes it into a separate branch and creates a pull or a merge request to the main one. A team lead reviews a code quality and efficiency. If something goes wrong, a team lead leaves comments on what to rewrite. If everything is okay, he or she merges a branch into the master branch, and the task is complete.

Sprint Retrospective

At the end of each sprint, we conduct sprint retrospective. During sprint retrospective, we evaluate what tasks we completed and what items we did not manage to achieve. We also take into account other indicators to assess productivity. We discuss what we did right and what went wrong, and consider any output to identify what we need to do, adapt, or improve to avoid past mistakes when planning the upcoming sprint.



GitHub

  **gibson042** approved these changes 23 days ago [View changes](#)

The concept seems reasonable. 🍌

```
src/ajax/script.js
48 48      var script, callback;
49 49      return {
50 50          send: function( _, complete ) {
51 51              script = jQuery( "<script>" ).prop( {
52 52                  charset: s.scriptCharset,
53 53                  src: s.url
54 54              } ).on(
54 54              ) .attr( s.scriptAttrs || {} ).on(
```

 **gibson042** 23 days ago **Member**
There should probably be a line break here.

src/merchant-orders/merchant-orders.module.ts **ADDED** [Side-by-side diff](#) [View file](#) [Comment](#) [...](#)

```
1 *import { Module } from '@nestjs/common';
2 *import { MerchantOrdersService } from '../merchant-orders.service';
3 *import { MongooseModule } from '@nestjs/mongoose';
4 *import { MerchantOrdersSchema } from '../merchant-orders.schema';
5 *import { MerchantOrdersRepository } from '../merchant-orders.repository';
6 +
7 *@Module({
8 +   imports: [
9 +     MongooseModule.forFeature([
10 +       { name: 'MerchantOrder', schema: MerchantOrdersSchema },
11 +     ]),
12 +   ],
13 +   providers: [MerchantOrdersService, MerchantOrdersRepository],
14 +   exports: [MerchantOrdersService],
15 + })
16 *export class MerchantOrdersModule {}
```

9 At Artsy, we use server-side rendering (SSR) to deliver every page you hit on artsy.net(https://artsy.net). We
10 decided on using SSR for many reasons, amongst them performance. (TODO: Maybe more about why we chose SSR, but I

 **ashfurrow** 4 days ago **Member** [+2](#) [🔍](#) [...](#)

Maybe more about why we chose SSR

Agreed, this'd be great to cover 🍌

Reply...

Resolve conversation

11 don't know much about that history.)
12

To Do

Data Base creation



rbac functionality for all
pages



order action create order
logic for existing ...



Doing

general bug fixes



Done

change existing schemas
to match actual or...



saved places action
service implementation



Quality Assurance

QA/QC engineers are responsible for maintaining product quality, regardless of the software development life cycle. It means that they come into play throughout the entire software development process. Quality assurance engineers review functions as they are performed, and introduce end-to-end features/acceptance tests to ensure that your end users receive a bug-free product that runs smoothly. QA/QC engineers are responsible for the user flow integrity and record tests to reduce the risk of broken features or missing components in user flows/user stories. The testing process is iterative, as once one defect is fixed, it can disclose other hidden ones or even create new bugs. If the task passes quality assurance, then it is sent to the project manager for review. If the task fails quality assurance, then it is sent back to the developer for further revision and fixing.

Quality assurance covers:

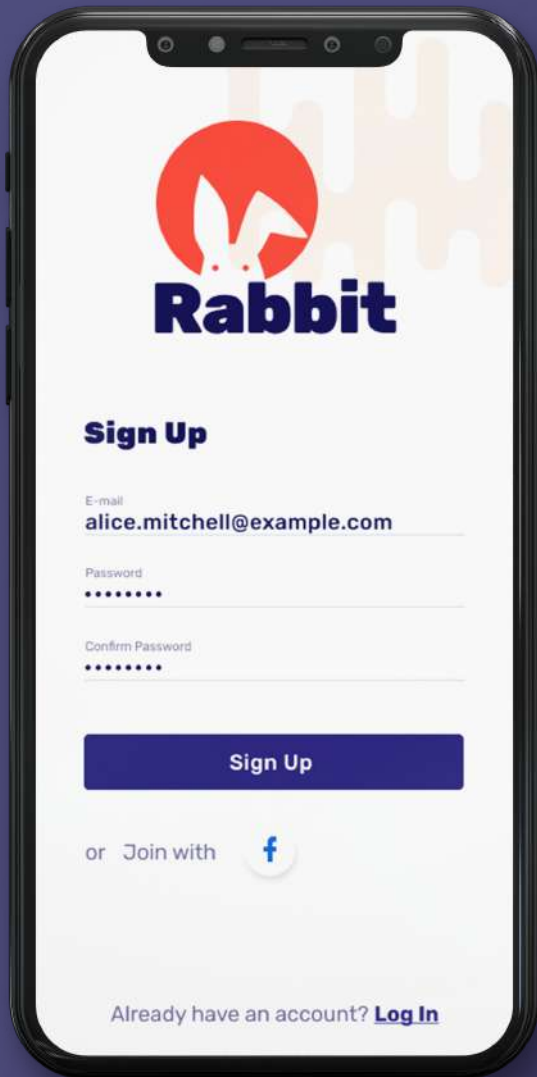
- *clarifying requirements*
- *test planning*
- *risk assessment methodology*
- *performance testing*
- *bug tracking*
- *bug reporting*
- *test automation*
- *release certification*

Version Release

After completing each sprint, we roll out a version release with limited working functionality for you to test, provide feedback, and ensure we do things right.

In case of mobile app development, upload build to TestFlight (iOS), APK (Android), or we can also send the file. In case of web-development, we deploy to staging.

Releases of each version can be carried out both before the product launch, and after.



v. 0.1.0

Launch

Once all the version releases have been completed and approved by you, your product is ready for beta testing. After end users beta test your app version, we roll out the public release.

Our DevOps engineers are responsible for making your product available to end users, ensuring its smooth performance in various operating systems or platforms. They scale concepts as and when required and develop code logistics so that code from developers can migrate to the development environment. DevOps engineers deploy software to production and development environment, provide data schema analysis and data migration, to reduce the risk of product inaccessibility.

Post-Release Support, Maintenance, Continued Development

The software development process does not end with your app or website launch. After the public release, your app moves to post-release support, maintenance, or continued development.

The digital market is rapidly changing and requires you to respond and adapt to new requirements instantly. To meet innovations, both mobile and web solutions require continuous testing and bug-fixing, maintenance, support, regular software updates, new functionality adoption.

Optimizing your app, you reduce the risk of the system crashes, ensure reliable and stable performance, increase data security.

Our services include:



Recommendations on managing a product



Optimizing performance



App or website backup



Error tracking, fixing, and bug reporting



Updating versions



Upgrading frameworks regarding OS upgrades



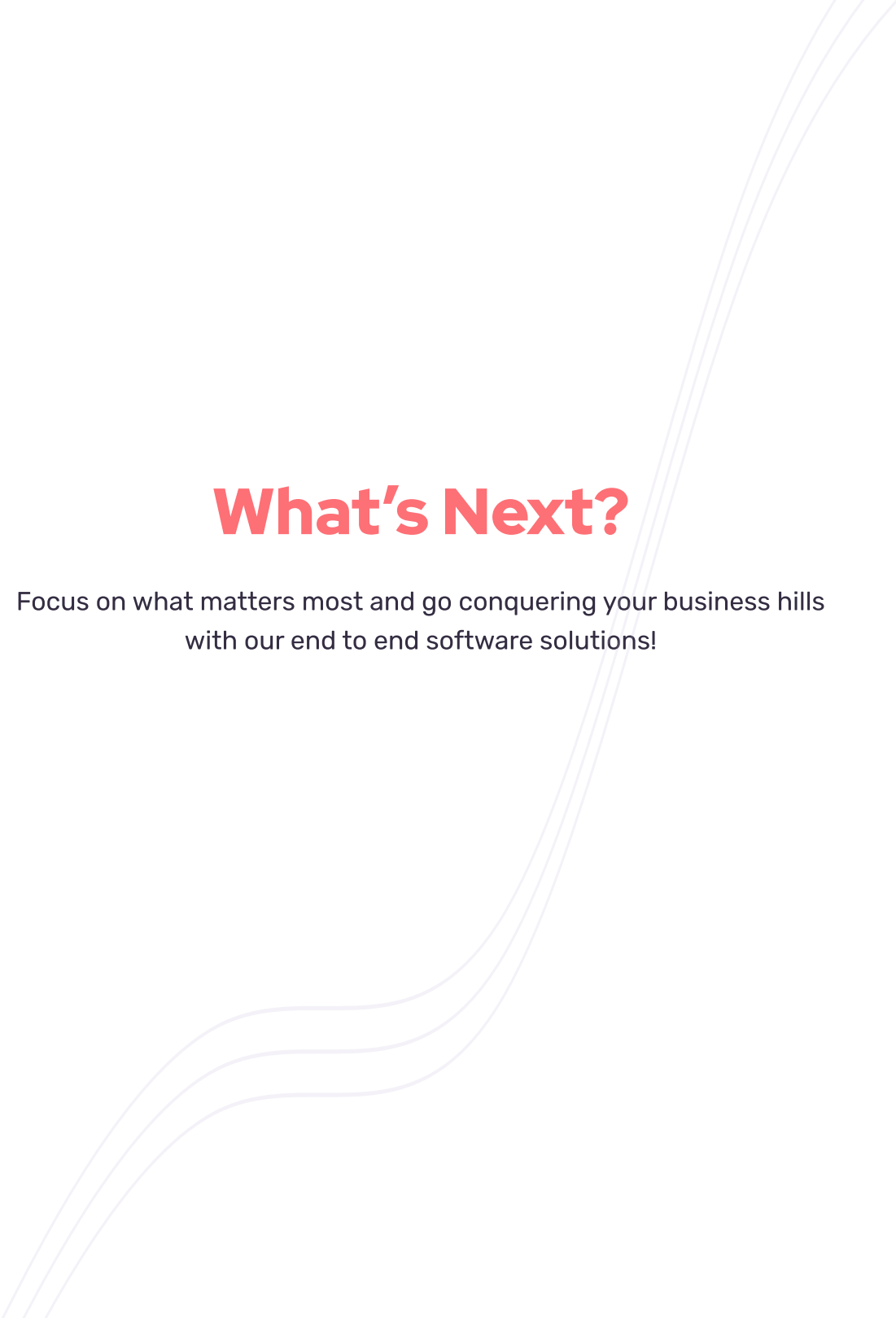
Managing integration dependencies



Adding new features



Migration to new OS



What's Next?

Focus on what matters most and go conquering your business hills
with our end to end software solutions!

For Those
Who Want
More

Contacts

ceo@opengeekslab.com

+ 46 700 931 035