

動画解説まとめ 30 / AI活用

AIでSNSの試作品を作る Antigravity入門

ログインして投稿できる小さなSNSを題材に、Antigravity・GitHub・Firebaseの役割、設定、エラー修正、機能追加までの実演を整理しました。

この動画の要点

プログラミング経験のある上田さんが、Google Antigravityへ日本語で依頼し、ログインと投稿ができるSNSの試作品を作る動画のまとめです。コードを作る場所、変更を保存する場所、サービスを動かす場所を分け、Firebaseの認証・データ保存・公開設定を進めます。実演中にはエラー修正とダークモード追加も行いました。最初から多機能化せず、動く小さな形を作って反応を見る考え方を中心に整理しています。

01 初めはログインと投稿だけに絞り、確認できる単位で機能を足す。

02 Antigravityで制作、GitHubで変更管理、Firebaseで実行という役割を理解する。

03 コードの生成と、認証・保存先・公開先の設定は別の作業として進める。

04 エラーは利用環境と状況を伝え、AIへの修正依頼に変換する。

05 動く試作品でアイデアを伝え、本格運用に向けた確認は別に行う。

資料の使い方

動画の要点を読み返し、紹介した手順を実践するときにお使いください。本文の時刻を押すと、元動画の該当箇所を開けます。

元動画：【超初心者向け！】未経験OK！AIでSNSを作ってお金を稼ぐ方法【Google Antigravity】【AIコーディング】

<https://www.youtube.com/watch?v=NejnUCnAOWw>

小さく動くSNSを、アイデアを試す材料にする

動画の該当箇所 1:31 / タップして開く

今回作るのは、大規模なSNSをそのまま再現する完成品ではなく、ログインして一件投稿できる試作品です。上田さんは、Webサービスを題材にすると、作りたいものを実際に動かして確認する練習がしやすいと説明しています。

SNSの面白さは、機能の多さだけで決まらないという話もあります。通知から短い時間で投稿する仕組みや、手書きだけでやり取りする案など、あえて制限を置くことで新しい楽しさが生まれる可能性を挙げています。これら是对談中の企画例で、今回すべてを実装したわけではありません。

最初の依頼を小さくする理由は、どこまで動いたかを確認でき、問題が起きたときに原因を絞りやすいためです。まずログイン・投稿を作り、その後にダークモードを追加する順番が実演されています。機能を一度に盛り込むより、動くことを確かめながら進めます。

- ・ 最初の完成物：ログインして短い文章を投稿できるSNS。
- ・ 次に足す機能：画面のダークモード切り替え。
- ・ 活用：企画の説明や、使ってもらった反応を確認するための試作品。

Antigravityへ、作るものを具体的に伝える

動画の該当箇所 4:05 / タップして開く

開発に使うのはAntigravityです。実演はMacで行われ、作業用の大きなフォルダーの下に、今回のSNS専用フォルダーを作って開きます。動画では「ニューTwitter」という名前を付けています。専用フォルダーの中に、AIが作るプログラムや関連ファイルをまとめます。

画面は左側がファイル一覧、中央が内容を見る部分、右側がAIへ指示する部分として説明されます。実演の依頼は、Firebaseを使い、ログインして投稿できる最低限のSNSを作ってほしい、という内容です。処理の途中で実行確認や変更の確認が入り、計画や作成されたファイルを確認しながら進みます。

AIが複数のファイルをまとめて作る点は便利ですが、漠然と修正を頼むと、意図しない場所まで変わる可能性も対談で指摘されています。特定の不具合や追加したい動作を言葉にすることが重要です。実演では処理ごとに待ち時間もあり、依頼した瞬間にすべて完成するわけではありません。

- ・ 一つのサービスに一つの作業フォルダーを用意する。
- ・ 使うサービスと最初に必要な動作を伝える。
- ・ AIの実行内容や変更を確認してから次へ進む。

制作・変更管理・実行の三つをつなぐ

動画の該当箇所 13:32 / タップして開く

動画では、Antigravityで作ったコードをGitHubに保存し、Firebaseとつないで動かす構成を採っています。Antigravityは制作する場所、GitHubはコードの変更や履歴を管理する場所、Firebaseは認証やデータ保存を含めてアプリを動かす場所、と役割を分けると理解しやすくなります。

GitHubへアカウントを用意し、今回のSNS用のリポジトリを作成します。実演では表示された案内のコマンドをコピーし、作業フォルダーからコードを送ります。後半には、Antigravityのソース管理画面からコミットしてプッシュする方法も紹介されています。どちらも、作った変更をGitHub側へ届ける操作です。

機能を作りかけの状態で公開先へ反映すると、利用者が動かない画面を見る可能性があります。そのため、通常は手元でテストし、うまく動く段階で変更を送るという説明があります。動画の初回実演は一部の手元確認を省いて公開先で試しており、その後実際に修正が必要になっています。

- ・ 手元で作る→動作を確かめる→変更を保存・送信する、という関係をつかむ。
- ・ 履歴があると、問題が出た変更を確認しやすくなる。
- ・ 自動反映を使う場合も、反映させる内容を先に確認する。

Firestoreに、ログインと保存先を用意する

動画の該当箇所 16:25 / タップして開く

Firebaseでプロジェクトを作り、Webアプリとして登録します。動画では同じSNS名で登録し、表示された設定値を開発側の設定ファイルへ対応させています。項目の数的一致しないところもあったため、必要な項目を見ながら入れる作業として紹介されています。具体的な値は各自のプロジェクトごとに異なります。

次に、投稿内容を管理するFirestore、写真などのファイルを保存するStorage、GoogleログインのためのAuthenticationを有効にします。三つは同じ保存機能ではなく、文章などのデータ、ファイル、利用者の認証をそれぞれ担います。画面上の名前を丸暗記するより、SNSのどの動作に必要なかと結び付けます。

実演ではテスト用の設定や、請求先・料金通知に関する設定も扱っています。少額を入力する場面は、設定した額をその場で支払う説明ではなく、料金の通知に関する設定として補足されています。この設定を費用の上限や自動停止の保証と読み替えず、実際の利用条件を確認する必要があります。

- ・ Firestore：投稿内容などのデータを扱う。
- ・ Storage：写真などのファイルを扱う。
- ・ Authentication：誰がログインしたかを扱う。
- ・ 動画内のテスト設定は、そのまま本格運用の完成設定とみなさない。

公開先と認証設定をそろえ、エラーを直す

動画の該当箇所 22:53 / タップして開く

公開にはFirebaseのApp Hostingを使います。GitHubと連携し、対象のリポジトリへのアクセスを許可して選択します。実演では反映元にmainを指定し、コードの自動反映を有効にしました。開発側に入れた設定値も公開側へ対応させ、公開処理を進めています。

公開用のドメインができれば、認証に使ってよいドメインの設定へ追加します。さらにGoogle Cloud側の同じプロジェクトで、OAuthに関係する画面、テストユーザー、JavaScriptの生成元、リダイレクト先などを確認しています。入力文字列の細部は字幕だけで確定できないため、動画の画面と動画で案内される手順を見て対応させる場面です。

一通り設定した後も、そのままでは動かない箇所がありました。実演者はAIへ状況を相談して修正依頼を考え、Antigravityで直しています。その後、Googleログインで自分の名前が表示され、文章を投稿できることを画面で確認しました。設定が終わったという返答と、目的の動作ができたという確認は分けて考えます。

- ・ 公開先の設定と、認証側で許可するドメインを対応させる。
- ・ エラー時は、自分の環境と起きた現象をAIへ伝える。
- ・ 最後はログインと投稿を実際に行って確認する。

一つ機能を足し、次の開発へつなげる

動画の該当箇所 30:24 / タップして開く

基本動作を確認した後、ヘッダー付近に切り替えボタンを置くダークモードを追加します。Antigravityへ希望を伝え、できた変更をソース管理からGitHubへ送り、公開先へ反映するのを待ちます。実演では手動のロールアウトも使い、最後に切り替えが動くことを確認しています。

この流れの価値は、機能の名称を知るだけでなく、依頼・変更・反映・確認の一周を経験できることです。動く試作品があれば、企画書だけよりも「こういうものを作りたい」と人へ伝えやすくなります。反応がよければ、自分で学んで拡張するか、専門家へ現物を見せて相談する選択肢があります。

対談では、OSや画面の更新で手順が少しずつ変わるため、自分はWindowsなのかMacなのか、どこが分からないのかをAIへ説明する習慣も勧めています。今回確認されたのは小規模な試作の動作であり、大人数への提供や収益化が完成した事例ではありません。利用者が増える段階では、費用・安定性・認証やデータの扱いを含めて運用を考えます。

- ・ 追加機能も、一つずつ作って反映結果を確かめる。
- ・ 分からない言葉は、自分の環境と経験を添えて説明を求める。
- ・ 試作品で反応を見てから、本格化に必要な作業を検討する。

実践の進め方

01 試作の範囲を決める

ログインと一件の投稿という最低限の動作に絞り、今回専用のフォルダーを開きます。

動画 5:17

02 SNSの基本部分を作成する

AntigravityへFirebaseを使うことと必要な動作を伝え、計画・実行・変更内容を確認します。

動画 7:30

03 コードと実行先を準備する

GitHubにコードを保存し、FirebaseへWebアプリを登録します。認証、投稿データ、ファイル保存の設定を分けて進めます。

動画 13:32

04 公開先で基本動作を確かめる

公開と認証の設定を対応させ、ログイン・投稿を実際に試します。動かない箇所は環境と症状をAIへ伝えて修正します。

動画 28:31

05 機能を一つ追加して再確認する

動画の例ではダークモードを依頼し、変更を送信・反映した後、切り替えを操作して確認しています。

動画 30:24

AIへの相談例

編集部が作成した活用例です。動画内の発言をそのまま書き起こしたものではありません。

【編集部による活用例・動画の逐語プロンプトではありません】この資料を根拠に、記載外の補足は区別し、私の状況を質問してから進めるようにしてください。使っているパソコン、作りたいSNS、プログラミング経験を確認し、最初に試す最低限の機能を整理してください。Antigravity・GitHub・Firebaseの役割を私の案に対応させ、試作品として確認する動作と、本格運用前に別途確かめる事項を分けてください。

資料について

- ・元動画の自動字幕から、試作の流れを整理しています。設定値やコマンドを転記した完全な操作手順書ではありません。画面の細部は元動画と利用時の案内を確認してください。
- ・料金、無料範囲、設定画面は動画時点の説明です。試作で動いたことは、費用・安全性・大規模運用・収益を保証するものではありません。