

Zane Carlson

UI

(Partial) ID Card Page

Password Recovery Page

(Partial) Credit Card Page

Auto Lock

Auto clear clipboard

Patterns

(CoR) Password Recovery Handler
(Singleton) Session Manager

Jared Wensley

Backend

Database Setup

Database handler class

Database login

Patterns

(Observer) Weak password
(Observer) Card Expiry

Michael Gharbieh

UI

Login Page

Main Page

Website Page

(Partial) Credit Card Page

(Partial) ID Card Page

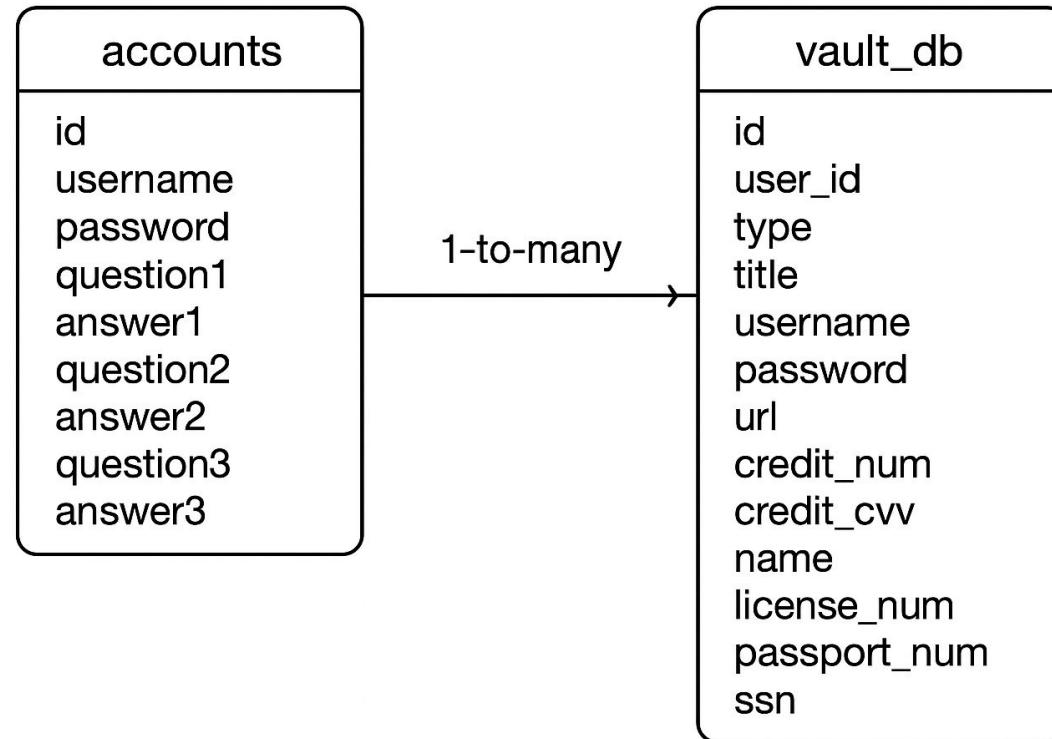
Secure Note Page

Light/Dark Mode

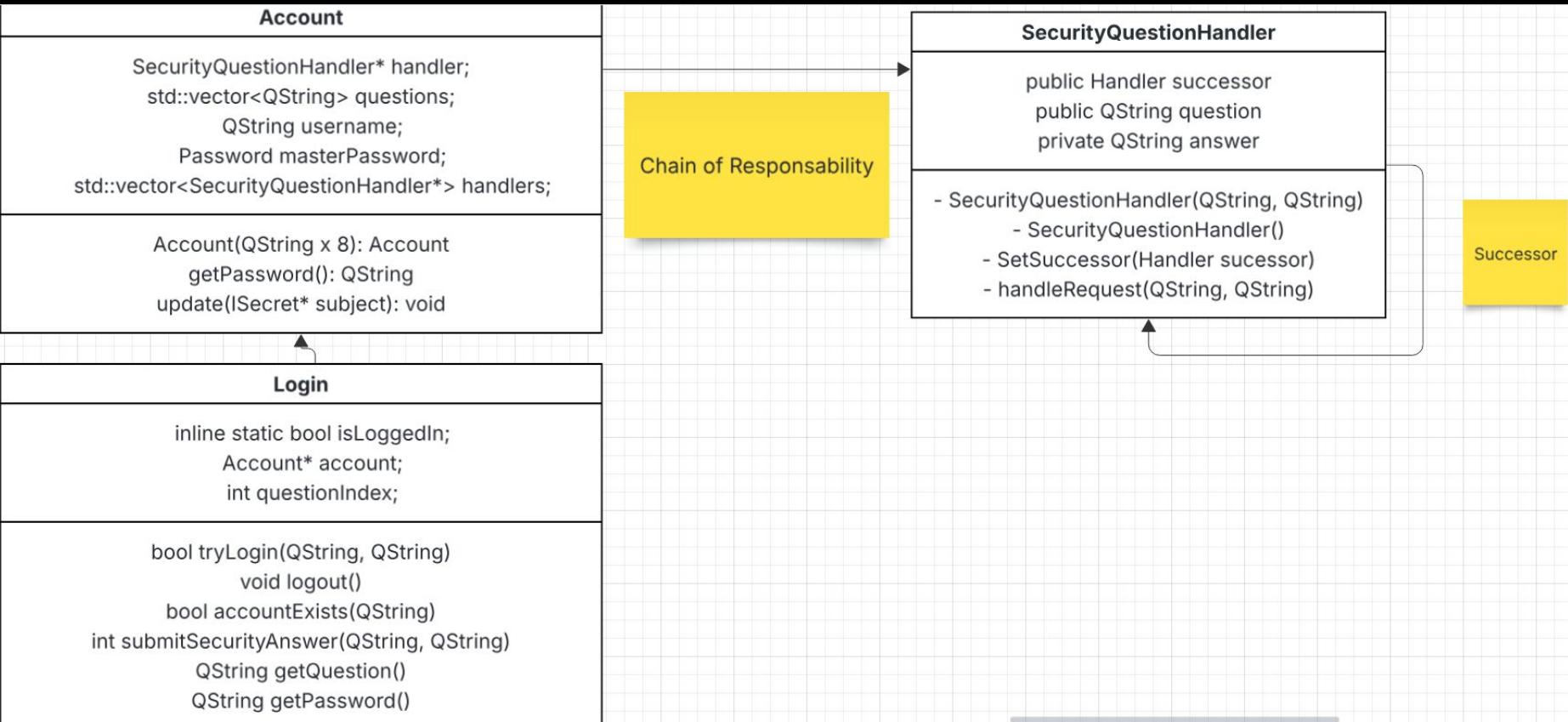
Patterns

(Builder) Password Generator

Database Schema



(CoR) Password Recovery Handler



Security Question Handler

```
#ifndef SECURITYQUESTIONHANDLER_H
#define SECURITYQUESTIONHANDLER_H

#include <QString> ❶ In included file: "Qt requires a C++17 compiler"

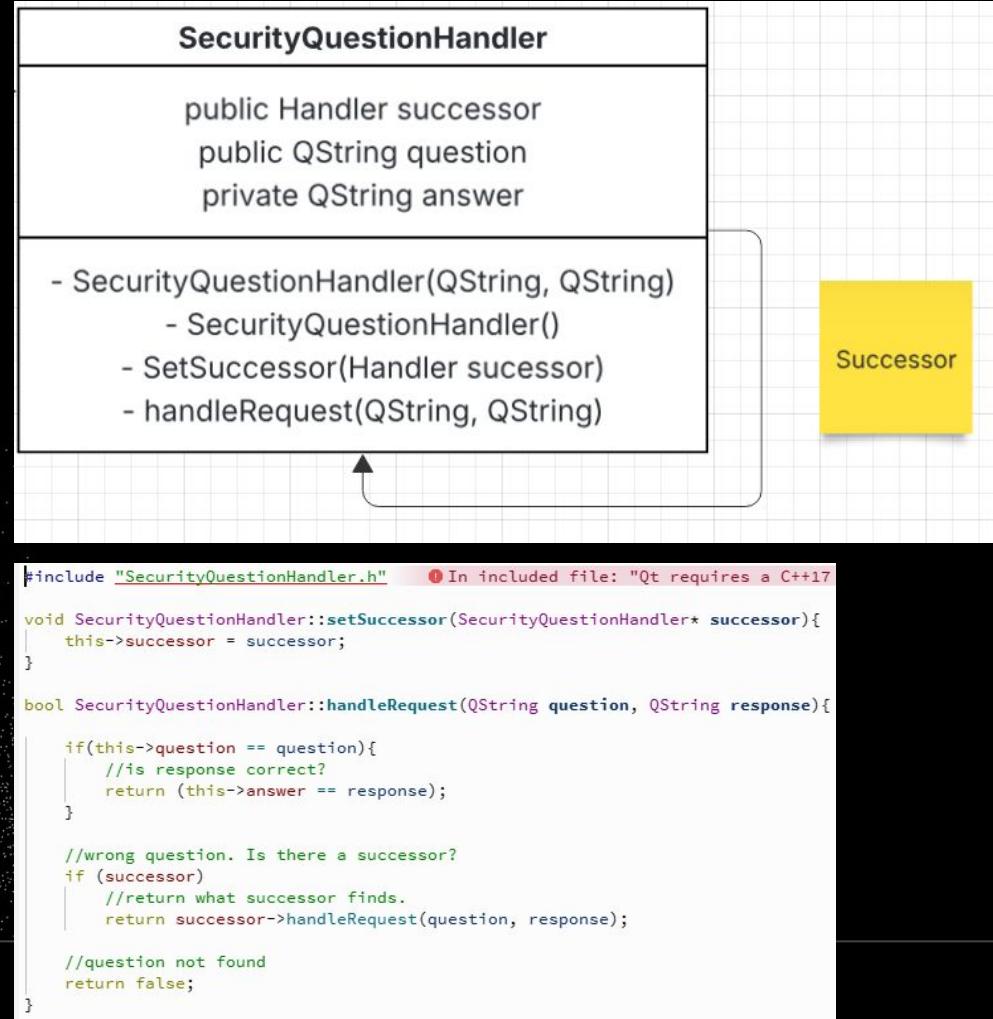
class SecurityQuestionHandler{
public:
    SecurityQuestionHandler* successor = nullptr;
    QString question;
    void setSuccessor(SecurityQuestionHandler* );
    bool handleRequest(QString question, QString Response);

    SecurityQuestionHandler(){
    };

    SecurityQuestionHandler(QString question, QString answer){
        this->question = question;
        this->answer = answer;
    }

private:
    QString answer;
};

#endif // SECURITYQUESTIONHANDLER_H
```



Account

```
class Account : public QObject, public IObserver {
    Q_OBJECT

public:
    SecurityQuestionHandler* handler;
    std::vector<QString> questions;

    void update(ISecret* subject) override {}

    Account(){}
    ~Account(){}

    Account(QString username, QString password,
            QString q1, QString a1,
            QString q2, QString a2,
            QString q3, QString a3){

        handlers.push_back(new SecurityQuestionHandler(q1, a1));
        handlers.push_back(new SecurityQuestionHandler(q2, a2));
        handlers.push_back(new SecurityQuestionHandler(q3, a3));

        for(int i = 0; i < handlers.size() - 1; i++){
            handlers[i]->setSuccessor(handlers[i+1]);
        }

        questions.push_back(q1);
        questions.push_back(q2);
        questions.push_back(q3);

        this->handler = handlers[0];

        this->username = username;
        this->masterPassword.setPass(password);
    }

    QString getPassword(){
        return masterPassword.getPassword();
    }

private:
    QString username;
    Password masterPassword;
    std::vector<SecurityQuestionHandler*> handlers;
};

#endif // ACCOUNT_H
```

Account

SecurityQuestionHandler* handler;
std::vector<QString> questions;
QString username;
Password masterPassword;
std::vector<SecurityQuestionHandler*> handlers;

Account(QString x 8): Account
getPassword(): QString
update(ISecret* subject): void

Login

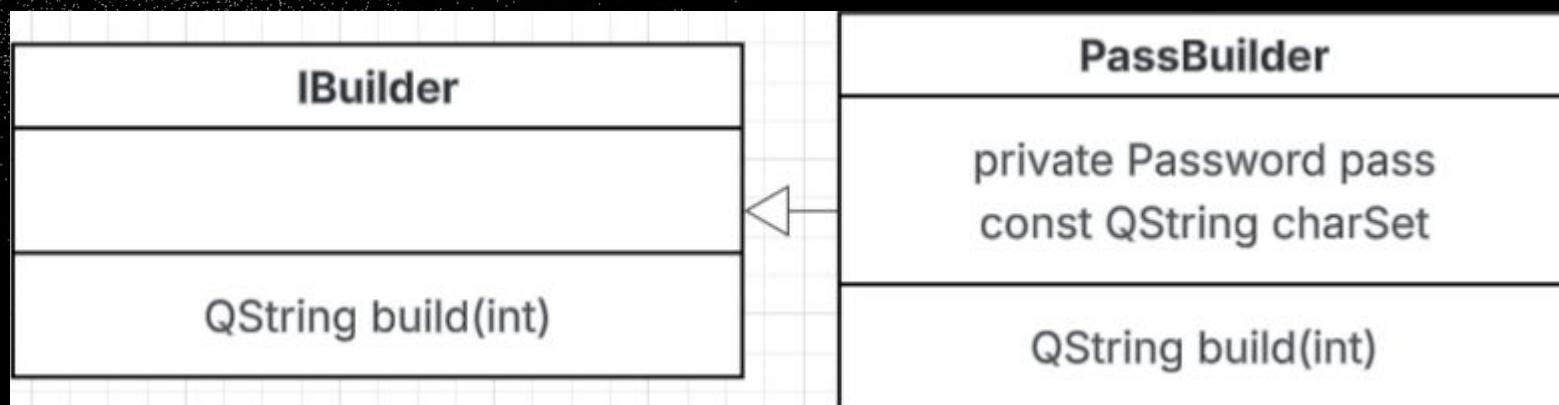
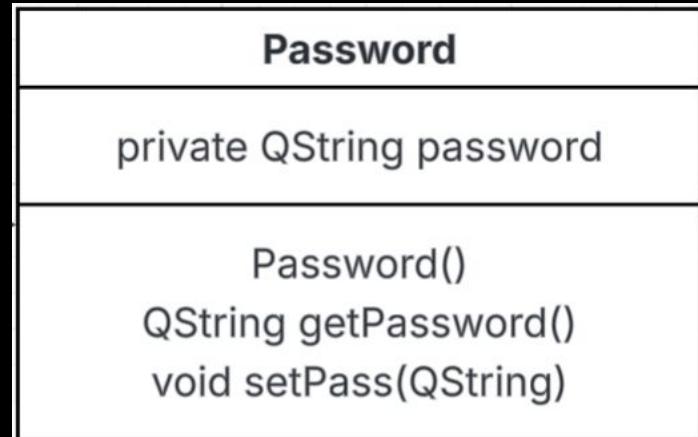
```
Q_INVOKABLE bool accountExists(QString username){  
    QSqlDatabase db = QSqlDatabase::database();  
    if (!db.isOpen()) {  
        qWarning() << "DB not open in trySecurityQuestions"; // In template: no member  
        return false;  
    }  
  
    QSqlQuery query;  
  
    query.prepare(  
        "SELECT password, question1, answer1, question2, answer2, question3, answer3 "  
        "FROM accounts "  
        "WHERE username = ?"  
    );  
    query.addValue(username);  
  
    if (!query.exec()) {  
        qWarning() << "Security questions query failed"; // In template: no member  
        // << query.lastError().text();  
        return false;  
    }  
  
    if (!query.next()) {  
        // Username not found  
        qDebug() << "No matching username:" << username; // In template: no member  
        return false;  
    }  
  
    QString password = query.value(0).toString();  
    QString q1= query.value(1).toString();  
    QString a1= query.value(2).toString();  
    QString q2= query.value(3).toString();  
    QString a2= query.value(4).toString();  
    QString q3= query.value(5).toString();  
    QString a3= query.value(6).toString();  
  
    // Delete previous account if any  
  
    account = new Account(username, password, q1, a1, q2, a2, q3, a3);  
    return true;  
}
```

```
Q_INVOKABLE int submitSecurityAnswer(QString question, QString response){  
    if(questionIndex >= 3) questionIndex = 0;  
    bool result = account->handler->handleRequest(question, response);  
    if(result){  
        questionIndex++;  
    }  
    return questionIndex;  
}  
  
Q_INVOKABLE QString getQuestion(int index) {  
    if (account == nullptr) return "invalid account";  
    if (account && index < account->questions.size())  
        return account->questions[index];  
    return "";  
}  
  
Q_INVOKABLE QString getPassword(){  
    return account->getPassword();  
}
```

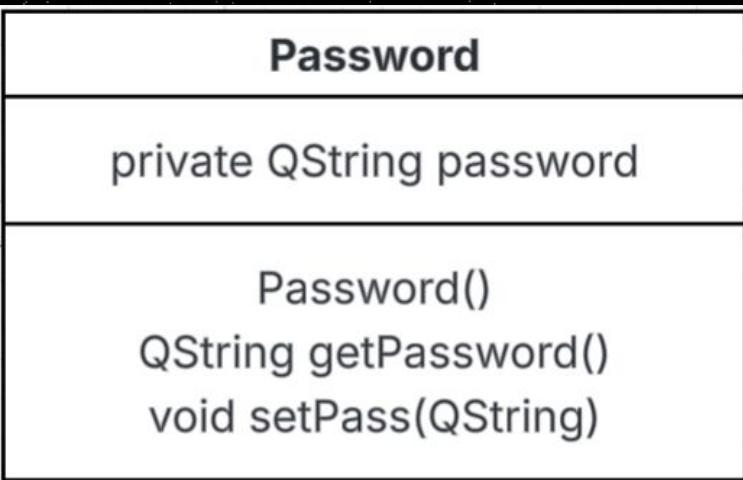
Login
inline static bool isLoggedIn; Account* account; int questionIndex; bool tryLogin(QString, QString) void logout() bool accountExists(QString) int submitSecurityAnswer(QString, QString) QString getQuestion() QString getPassword()

```
Button {  
    id: submitButton  
    text: "Submit Answer"  
    onClicked: { inactiveTimer.restart()  
        var questionIndex = LOGIN.submitSecurityAnswer(questionText.text, answerField.text)  
        if(questionIndex === currentQuestionIndex){  
            statusText.text = "Incorrect, try again"  
        }  
        else{  
            statusText.text = ""  
            answerField.text = ""  
        }  
        currentQuestionIndex = questionIndex  
  
        questionText.text = LOGIN.getQuestion(questionIndex)  
        if(questionIndex >= numSecurityQuestions){  
            currentQuestionIndex = 0  
            questionText.text = "MASTER PASSWORD: " + LOGIN.getPassword()  
            submitButton.enabled = false  
        }  
    }  
}
```

(Builder) Password Generator



Password



```
class Password : public QObject\n{\n    Q_OBJECT\n\nprivate:\n    QString password;\n\npublic:\n    Password() {}\n    QString getPassword() {return password;}\n    void setPass(QString p) {password = p;}\n};
```

PassBuilder

```
class PassBuilder : public IBuilder
{
    Q_OBJECT
    QML_ELEMENT
private:
    Password pass;
    const QString charSet = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789@#$%^&*()_+=[]{}|;:,.<>?";

public:
    Q_INVOKABLE QString build(int len) override
    {
        std::default_random_engine generator(std::chrono::system_clock::now().time_since_epoch().count());
        std::uniform_int_distribution<int> distribution(0, charSet.length() - 1);

        QString generatedPass = "";
        for(int i = 0; i < len; i++)
        {
            generatedPass += charSet[distribution(generator)];
        }

        pass.setPass(generatedPass);
        return pass.getPassword();
    }
};
```

PassBuilder

private Password pass
const QString charSet

QString build(int)

UI Interaction

Add Website

Website:

Username:

Password:

[Use auto-generated password](#)

```
Button {
    id: pwdBuilderButton
    width: 175
    height: 30
    anchors{
        top: parent.top
        left: parent.left
        right: parent.right
    }

    font{
        pixelSize: 18
        underline: true
    }

    contentItem: Text {
        id: pwdBuilderText
        text: "Use auto-generated password"
        font: parent.font
        color: "#0081FF"
        anchors.centerIn: parent
    }

    background: Rectangle {
        color: "transparent"
        radius: 10
    }

    HoverHandler {
        cursorShape: Qt.PointingHandCursor
    }

    onClicked: { inactiveTimer.restart()
        websitePassInput.text = PASSBUILDER.build(10) //10 chars long, can be changed
        passInputBackground.border.color = "#969696"
    }
}
```

Observer Pattern

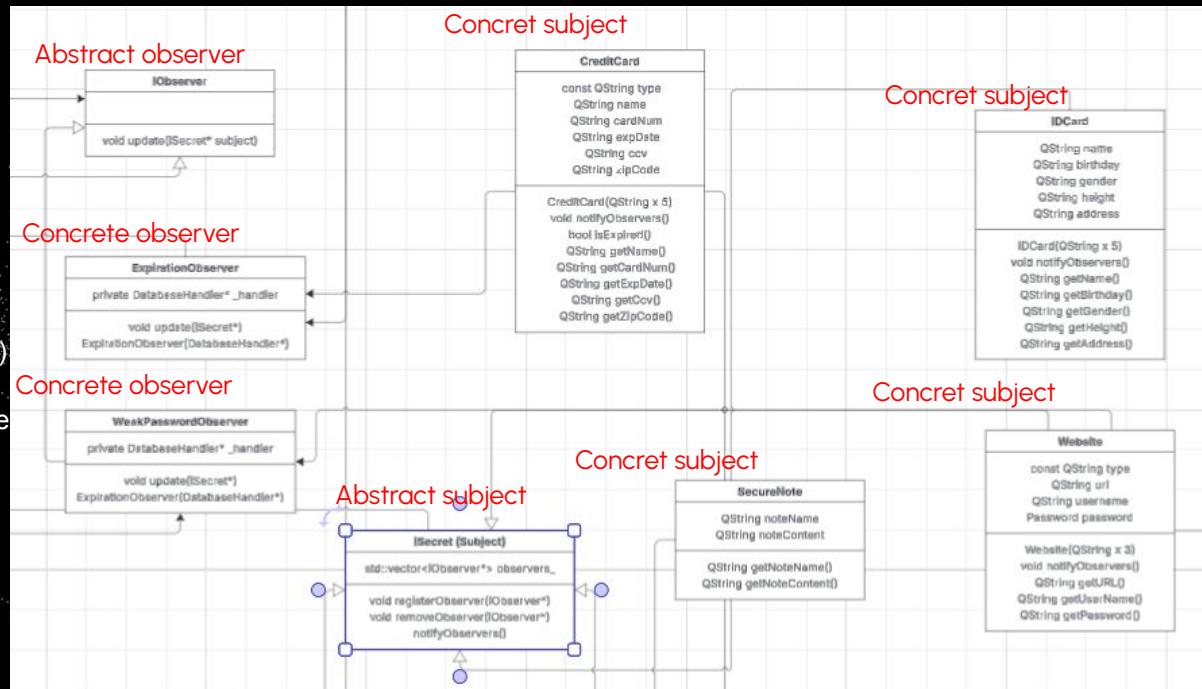
QML signal

databaseHandler::save...

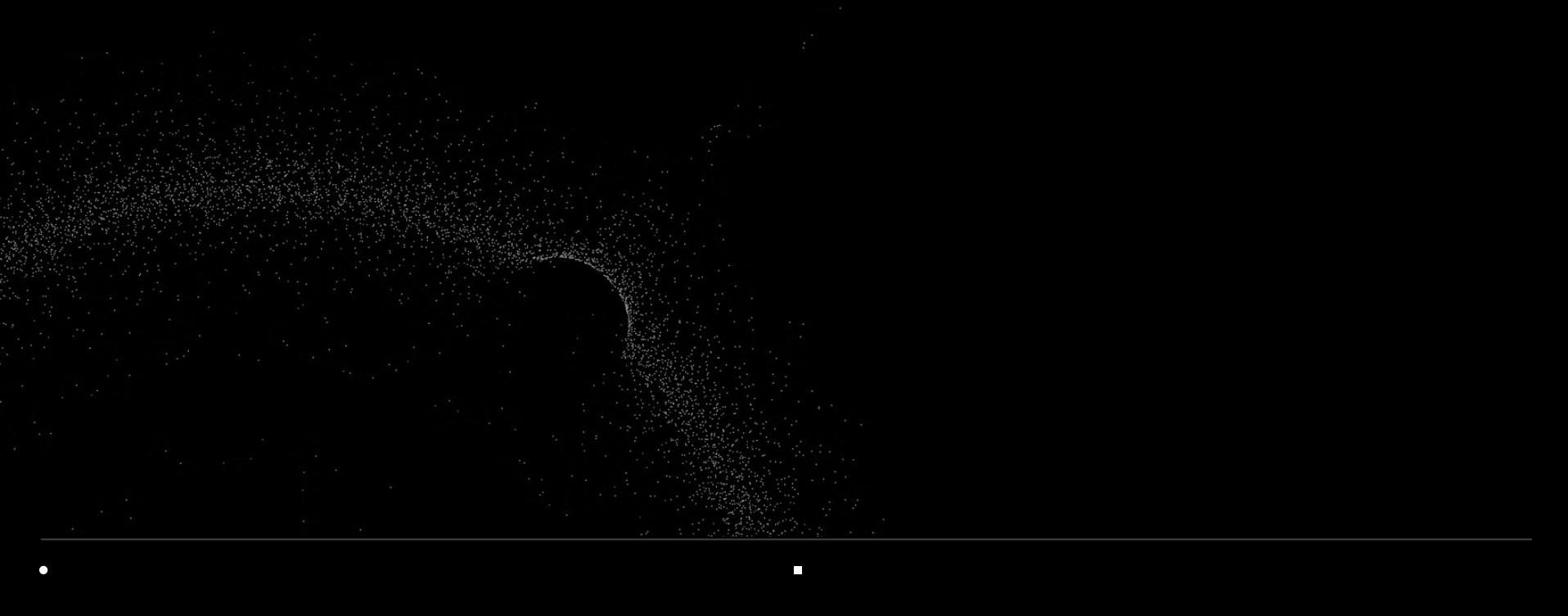
Website::notifyObservers

weakPasswordObserver::update(ISecret* subject)

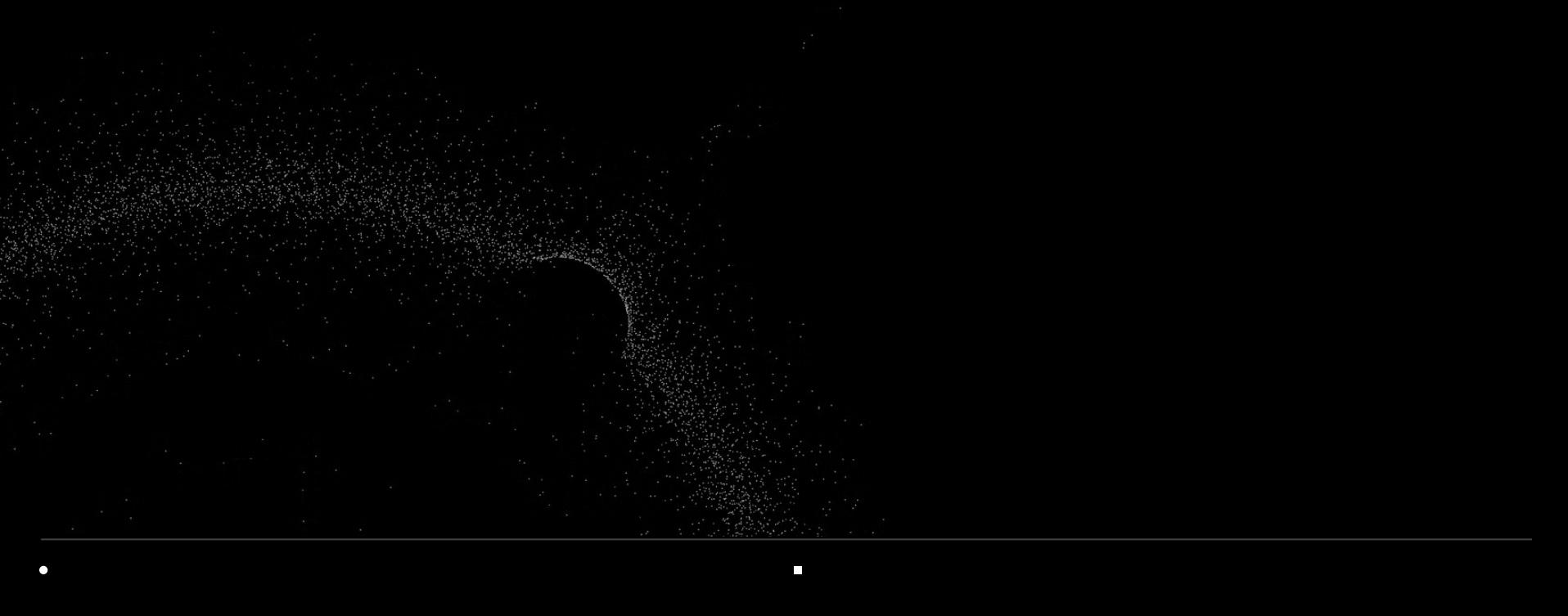
DatabaseHandler::ReportWeakpassword finds the
index in the vault emits for qml



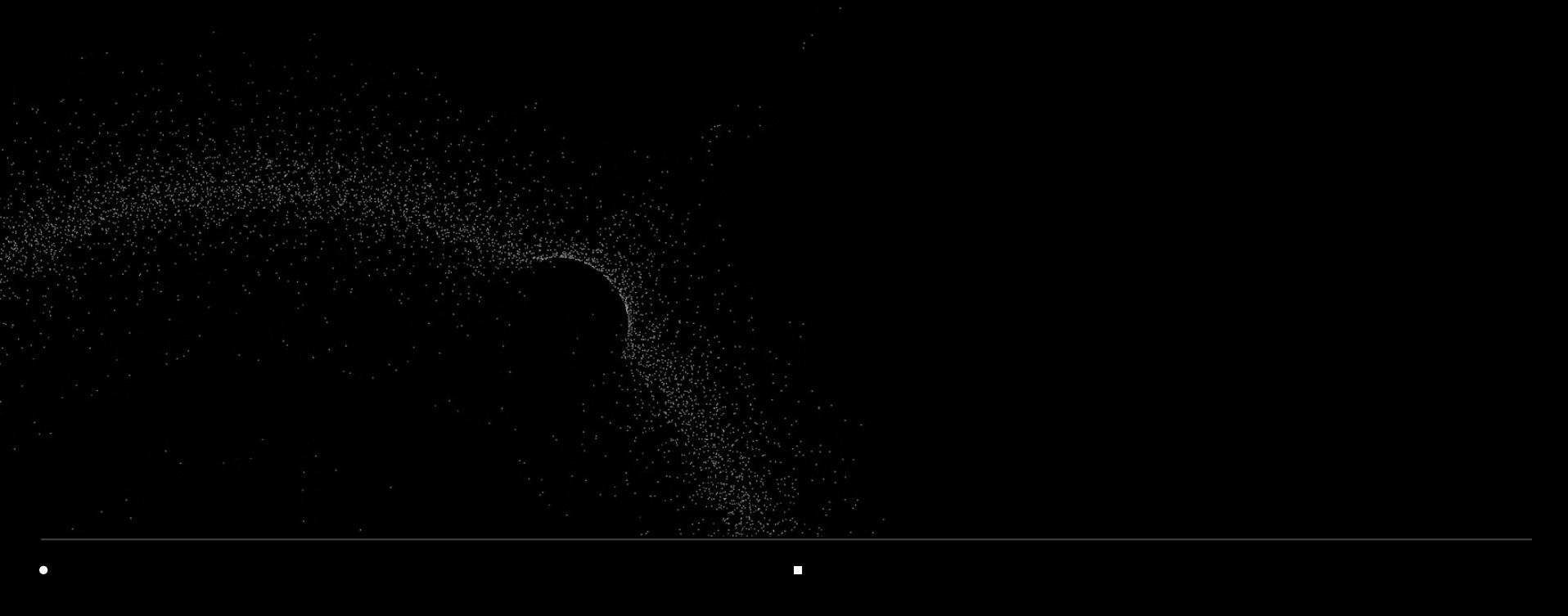
UI Interaction



UI Interaction



UI Interaction



Meet the team



Full Name
Title



Full Name
Title



Full Name
Title



Full Name
Title



Full Name
Title



Full Name
Title



Full Name

Title

Write a brief bio mentioning their background, area of expertise, and what their role in this partnership will be.



Full Name

Title

Write a brief bio mentioning their background, area of expertise, and what their role in this partnership will be.



Full Name

Title

Write a brief bio mentioning their background, area of expertise, and what their role in this partnership will be.



Full Name

Title

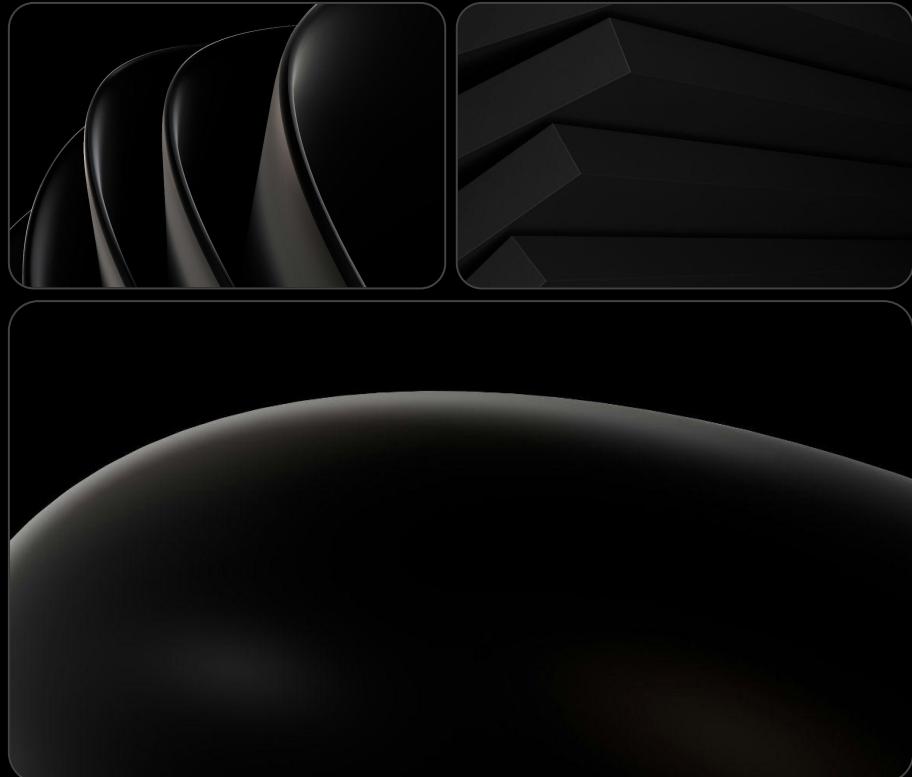
Write a brief bio mentioning their background, area of expertise, and what their role in this partnership will be.

Write a bold, compelling statement about what the company wants to achieve.

Write a bold, compelling statement about what the company wants to achieve.

Explain how a partnership would help make this goal a reality and why it's worth pursuing together. Think about how your potential partner can contribute.

Learn more



What we do



Describe your company's products and services.
Keep your potential partner in mind. What do you offer that's relevant to their business?

Our accomplishments

Highlight your company's growth, metrics, awards, and achievements.



Industry award
Product or campaign

00%

Market share

#00

Rank in the industry

Certifications



"Quote from published media coverage about your company"

Link to article

Write a statement about the core principles that guide your company's actions.

1



2



3



Add a value or belief

Define this value and explain how it reflects your company's culture or business aspirations.

Add a value or belief

Examples of company values or beliefs might include teamwork, innovation, or customer focus.

Add a value or belief

For each value, describe how it makes your company desirable as a business partner.

Customer segment title

"Summarize your key values, as if you were speaking directly to your customer segment."



Introduce your customer segment. Include demographic information, such as age range or location. Mention their needs, aspirations, and pain points.

Explain how your product or service solves these pain points and realizes your customers' goals.

Our customers



Customer segment title

Age range: 00-00

Education level: Highest education

Status: Marital status

Location: City or state

Archetype: Tech-savvy

Needs and motivations

- What does this segment want?
- What motivates them?
- What kind of products or services are they looking for?

Pain points

- What interferes with their needs, goals, and motivations?
- What frustrates them in their daily life?

Favorite channels

Logo

Logo

Logo

Technical skills

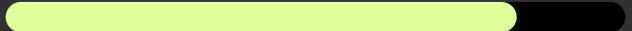
Device 1



Device 2



Device 3



Purchasing habits



Online store

Social media

Physical store

Our successful partnerships

Celebrate what your company achieved with this partner.

Introduce one of your current partners. Mention their industry or sector, then describe what you accomplished together. Include key initiatives and outcomes.

Learn more



Our successful partnerships

Partnership 1

Quarter, Year

Introduce one of your current partners. Mention their industry or sector, then describe what you accomplished together. Include key initiatives and outcomes.

Learn more



Partnership 2

Quarter, Year

Introduce one of your current partners. Mention their industry or sector, then describe what you accomplished together. Include key initiatives and outcomes.

Learn more



Our successful partnerships

Partnership 1

Quarter, Year

LOGO

Introduce one of your current partners. Mention their industry or sector, then describe what you accomplished together. Include key initiatives and outcomes.

Partnership 2

Quarter, Year

LOGO

Introduce one of your current partners. Mention their industry or sector, then describe what you accomplished together. Include key initiatives and outcomes.

Partnership 3

Quarter, Year

LOGO

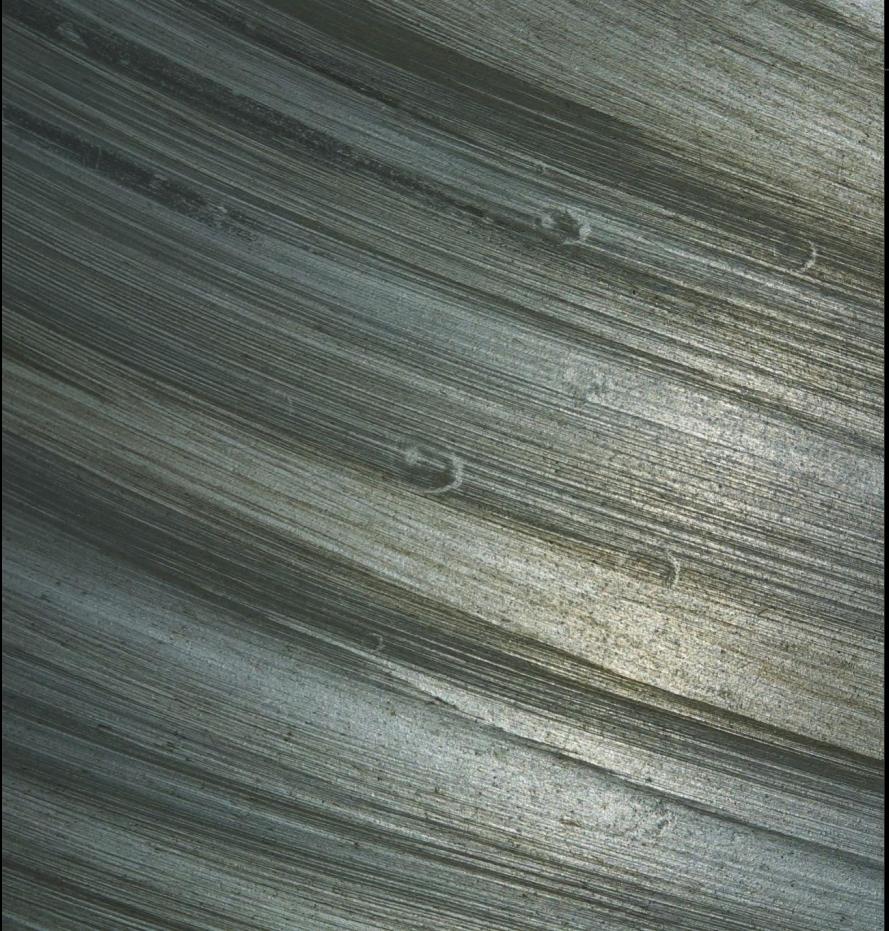
Introduce one of your current partners. Mention their industry or sector, then describe what you accomplished together. Include key initiatives and outcomes.

Partnership 4

Quarter, Year

LOGO

Introduce one of your current partners. Mention their industry or sector, then describe what you accomplished together. Include key initiatives and outcomes.

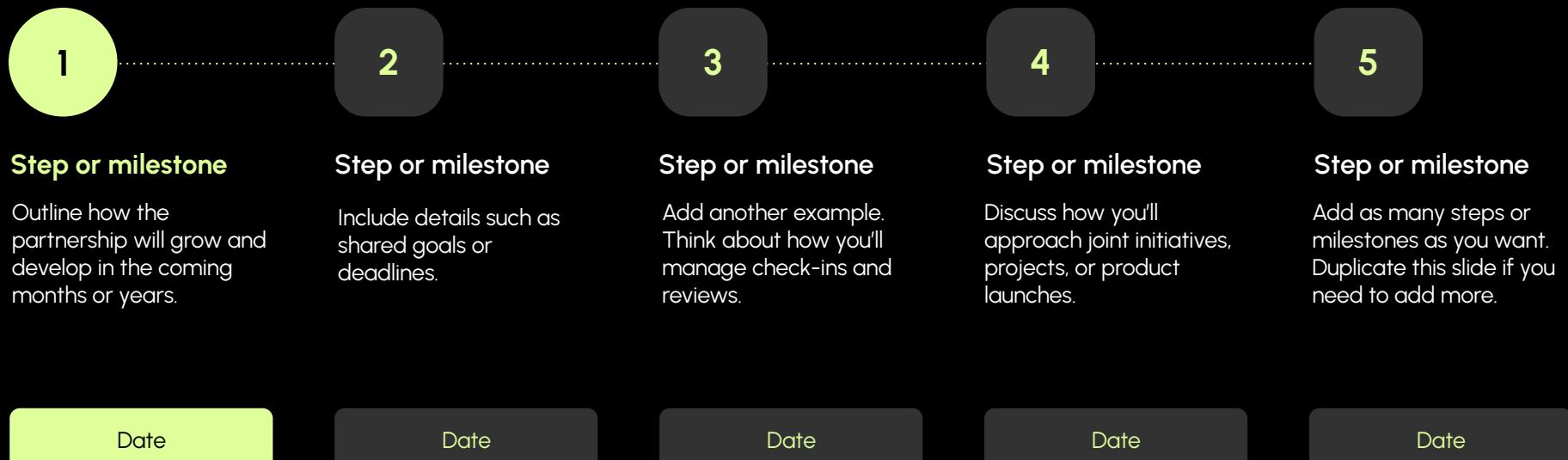


What we can do together

Write a statement about why you want to work with this partner.

Summarize why this partnership would be beneficial for both parties. Show how your values and goals align, and explain how the partnership could help you realise these goals.

How the partnership will work



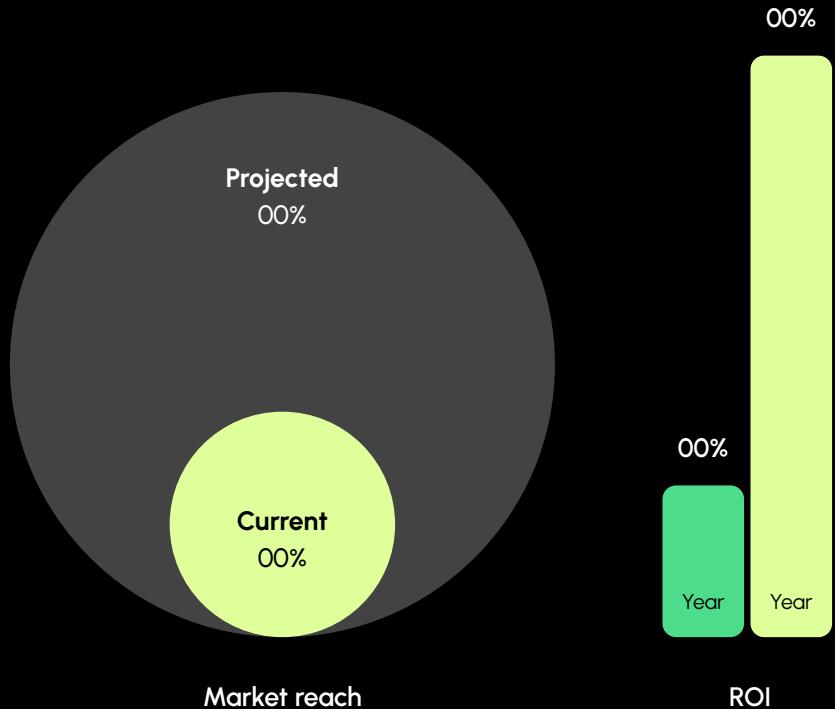
Let's work together

Invite your potential partner to join your business.

- Mention 3 or 4 partnership benefits
- Each benefit should build on the information outlined in the previous slides
- Include key performance indicators that support your proposal, such as the projected return on investment (ROI) for both parties



Invite your potential partner to join your business.



Demonstrate the benefits of the partnership through charts, graphs, and statistics. Use this space to explain what the data shows and how it impacts both businesses.

Timeline

1



Quarter, Year

Outline the next steps of the partnership plan.

2



Quarter, Year

Set a deadline for drafting a partnership agreement, for example.

3



Quarter, Year

Deliver the implementation plan for the partnership.

4



Quarter, Year

Allocate resources and decide on communication channels.

Thank you

Ready for what's next?
Let's talk

Full Name (of the point of contact)
Email
Phone number