

問4 コンテナ型仮想化技術に関する次の記述を読んで、設問に答えよ。

C社は、レストランの予約サービスを提供する会社である。C社のレストランの予約サービスを提供するWebアプリケーションソフトウェア（以下、Webアプリという）は、20名の開発者が在籍するWebアプリ開発部で開発、保守されている。C社のWebアプリにアクセスするURLは、“https://www.example.jp/”である。

Webアプリには、機能X、機能Y、機能Zの三つの機能があり、そのソースコードやコンパイル済みロードモジュールは、開発期間中に頻繁に更新されるので、バージョン管理システムを利用してバージョン管理している。また、Webアプリは、外部のベンダーが提供するミドルウェアA及びミドルウェアBを利用しており、各ミドルウェアには開発ベンダーから不定期にアップデートパッチ（以下、パッチという）が提供される。パッチが提供された場合、C社ではテスト環境で一定期間テストを行った後、顧客向けにサービスを提供する本番環境のミドルウェアにパッチを適用している。

このため、Webアプリの開発者は、本番環境に適用されるパッチにあわせて、自分の開発用PCの開発環境のミドルウェアにもパッチを適用する必要がある。開発環境へのパッチは、20台の開発用PC全てに適用する必要がある、作業工数が掛かる。

そこで、Webアプリ開発部では、Webアプリの動作に必要なソフトウェアをイメージファイルにまとめて配布することができるコンテナ型仮想化技術を用いて、パッチ適用済みのコンテナイメージを開発者の開発用PCに配布することで、開発環境へのミドルウェアのパッチ適用工数を削減することについて検討を開始した。コンテナ型仮想化技術を用いた開発環境の構築は、Webアプリ開発部のDさんが担当することになった。

[Webアプリのリリーススケジュール]

まずDさんは、今後のミドルウェアへのパッチ適用とWebアプリのリリーススケジュールを確認した。今後のリリーススケジュールを図1に示す。

リリース案件		説明	7月	8月	9月	10月	11月	12月
本番環境 へのパッチ 適用	ミドルウェアA パッチ適用	バージョン10.1.2 パッチの適用			テスト	▲リリース		
	ミドルウェアB パッチ適用	バージョン15.3.4 パッチの適用				テスト	▲リリース	
Webアプリ 開発	10月1日リリース 向け開発	機能Xの変更	設計	開発	テスト	▲リリース		
	11月1日リリース 向け開発	機能Yの変更		設計	開発	テスト	▲リリース	
	12月1日リリース 向け開発	機能Zの変更			設計	開発	テスト	▲リリース

図1 今後のリリーススケジュール

C社では、ミドルウェアの公開済みのパッチを計画的に本番環境に適用しており、本番環境のミドルウェアAのパッチ適用が10月中旬に、ミドルウェアBのパッチ適用が11月中旬に計画されている。また、10月、11月、及び12月に向けて三つのWebアプリ開発案件が並行して進められる予定である。開発者は各Webアプリ開発案件のリリーススケジュールを考慮し、リリース時点の本番環境のミドルウェアのバージョンと同一のバージョンのミドルウェアを開発環境にインストールして開発作業を行う必要がある。

なお、二つのミドルウェアでは、パッチ提供の場合にはバージョン番号が0.0.1ずつ上がることがミドルウェアの開発ベンダーから公表されている。また、バージョン番号を飛ばして本番環境のミドルウェアにパッチを適用することはない。

[コンテナ型仮想化技術の調査]

次にDさんは、コンテナ型仮想化技術について調査した。コンテナ型仮想化技術は、一つのOS上に独立したアプリケーションの動作環境を構成する技術であり、や上に仮想マシンを動作させるサーバ型仮想化技術と比較してが不要となり、CPUやメモリを効率良く利用できる。C社の開発環境で用いる場合には、Webアプリの開発に必要な指定バージョンのミドルウェアをコンテナイメージにまとめ、それを開発者に配布する。

[コンテナイメージの作成]

まずDさんは、基本的なライブラリを含むコンテナイメージをインターネット上の公開リポジトリからダウンロードし、Webアプリの開発に必要な二つのミドルウェア

の指定バージョンをコンテナ内にインストールした。次に、コンテナイメージを作成し社内リポジトリへ登録して、C社の開発者がダウンロードできるようにした。

なお、Web アプリのソースコードやロードモジュールは、バージョン管理システムを利用してバージョン管理し、①コンテナイメージに Web アプリのソースコードやロードモジュールは含めないことにした。Dさんが作成したコンテナイメージの一覧を表1に示す。

表1 Dさんが作成したコンテナイメージの一覧

コンテナイメージ名	説明	ミドルウェア A バージョン	ミドルウェア B バージョン
img-dev_oct	10月1日リリース向け開発用	(省略)	(省略)
img-dev_nov	11月1日リリース向け開発用	d	e
img-dev_dec	12月1日リリース向け開発用	10.1.2	15.3.4

〔コンテナイメージの利用〕

Web アプリ開発部の Eさんは、機能 Xの変更を行うために、Dさんが作成したコンテナイメージ“img-dev_oct”を社内リポジトリからダウンロードし、開発用 PCでコンテナを起動させた。Eさんが用いたコンテナの起動コマンドの引数（抜粋）を図2に示す。

```
-p 10443:443 -v /app/FuncX:/app img-dev_oct
```

図2 Eさんが用いたコンテナの起動コマンドの引数（抜粋）

図2中の -p オプションは、ホスト OS の 10443 番ポートをコンテナの 443 番ポートにバインドするオプションである。なお、コンテナ内では 443 番ポートで Web アプリへのアクセスを待ち受ける。さらに、-v オプションは、ホスト OS のディレクトリ“/app/FuncX”を、コンテナ内の“/app”にマウントするオプションである。

Eさんが Web アプリのテストを行う場合、開発用 PC のホスト OS で実行される Web ブラウザから②テスト用の URL へアクセスすることで“img-dev_oct”内で実行されている Web アプリにアクセスできる。また、コンテナ内に作成されたファイル“/app/test/test.txt”は、ホスト OS の f として作成される。

12月1日リリース向け開発案件をリリースした後の12月中旬に、10月1日リリース向け開発で変更を加えた機能Xに処理ロジックの誤りが検出された。この誤りを12月中旬に修正して本番環境へリリースするために、Eさんは③あるコンテナイメージを開発用PC上で起動させて、機能Xの誤りを修正した。

その後、Dさんはコンテナ型仮想化技術を活用した開発環境の構築を完了させ、開発者の開発環境へのパッチ適用作業を軽減した。

設問1 [コンテナ型仮想化技術の調査] について答えよ。

- (1) 本文中の ～ に入れる適切な字句を解答群の中から選び、記号で答えよ。

解答群

- | | |
|------------|----------|
| ア アプリケーション | イ ゲスト OS |
| ウ ハイパーバイザー | エ ホスト OS |
| オ ミドルウェア | |

- (2) 今回の開発で、サーバ型仮想化技術と比較したコンテナ型仮想化技術を用いるメリットとして、最も適切なものを解答群の中から選び、記号で答えよ。

解答群

- ア 開発者間で差異のない同一の開発環境を構築できる。
- イ 開発用PC内で複数Webアプリ開発案件用の開発環境を実行できる。
- ウ 開発用PCのOSバージョンに依存しない開発環境を構築できる。
- エ 配布するイメージファイルのサイズを小さくできる。

設問2 [コンテナイメージの作成] について答えよ。

- (1) 本文中の下線①について、なぜDさんはソースコードやロードモジュールについてはコンテナイメージに含めずに、バージョン管理システムを利用して管理するのか、20字以内で答えよ。

- (2) 表1中の , に入れる適切なミドルウェアのバージョンを答えよ。

設問3 [コンテナイメージの利用] について答えよ。

- (1) 本文中の下線②について、Webブラウザに入力するURLを解答群の中から選び、記号で答えよ。

解答群

ア `https://localhost/`

イ `https://localhost:10443/`

ウ `https://www.example.jp/`

エ `https://www.example.jp:10443/`

(2) 本文中の に入れる適切な字句を，パス名/ファイル名の形式で答えよ。

(3) 本文中の下線③について，起動するコンテナイメージ名を表 1 中の字句を用いて答えよ。